

Mensile di elettronica applicata, attualità scientifica, novità tecnologiche.

€ 5,00

Elettronica In

oltre l'elettronica

www.elettronica.in.it

Anno XIV - n. 133
Dicembre 2008 /
Gennaio 2009



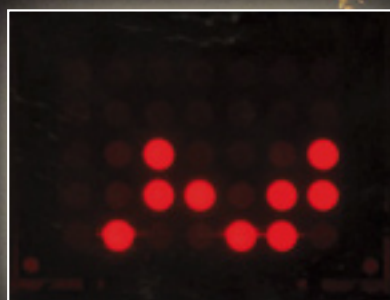
Tutto Robot, divertiti e impara

- Modulo speech miniatura
- Orologio binario
- Sistema di localizzazione web multiutente
- Dispositivo antiblackout domestico
- Localizzatore GPS/GSM miniatura, la realizzazione
- Cloud computing, dal PC al World Wide Computer

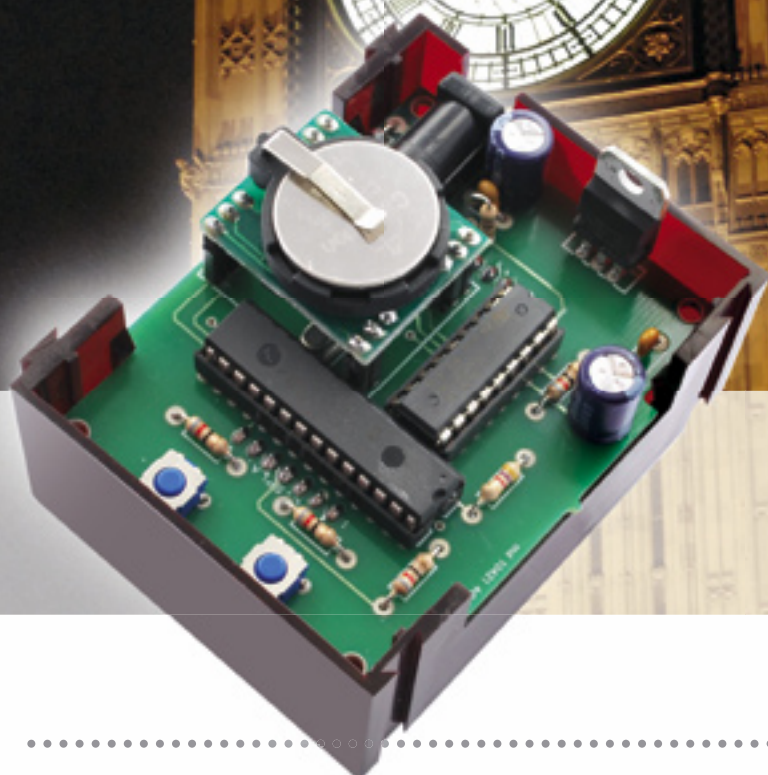
ROBOT BIPEDE



ESCLUSIVO!
Corso ZigBee



Anziché utilizzare i tradizionali quadranti con le lancette o i display numerici, visualizza l'ora mediante led, a ognuno dei quali è associato un valore binario. Ore, minuti e secondi sono espressi mediante sei file di led.

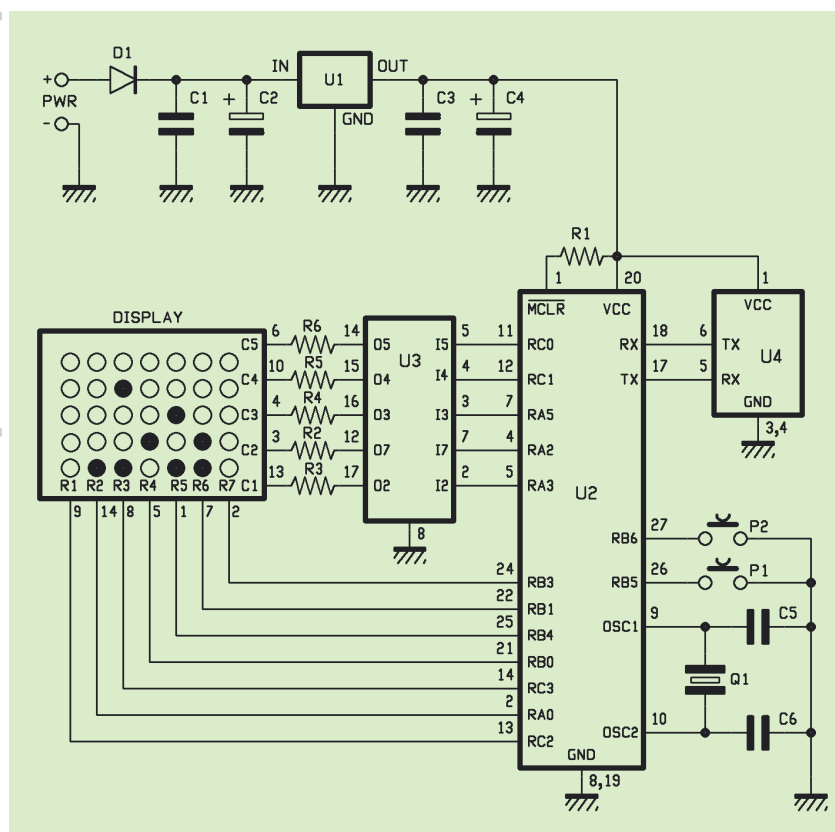


OROLOGIO BINARIO

di ARSENIO SPADONI

Fino a una trentina d'anni fa, quando si diceva orologio si intendeva quello a lancette mosse da un sistema meccanico alla base del quale c'era una molla, tesa manualmente (carica manuale) o mediante un volano (carica automatica) azionato dai normali movimenti del polso. Venne poi la moda degli orologi al quarzo e, in special modo, di quelli a display numerici, apprezzati sia perché erano un segno dello sviluppo esponenziale della tecnologia di quegli anni, sia per la più immediata leggibilità rispetto al sistema a lancette. Un fenomeno mai

apprezzato dai cultori dell'orologio di classe; ma anche qualcosa che, come tutte le mode, svanì nel giro di un decennio. Va però detto che da qualche tempo, per mano di alcuni nostalgici dei "diodi luminosi" e dei "cristalli liquidi" o di qualche stilista amante della tecnologia, sono comparsi orologi molto particolari, non certo fatti per essere letti ma, casomai, per giocare tra il mistero e la novità. Un po' come certe opere d'arte moderna, poco hanno dell'immediato, ed è forse per questo che chi vuole distinguersi ad ogni costo li apprezza. Parliamo dei



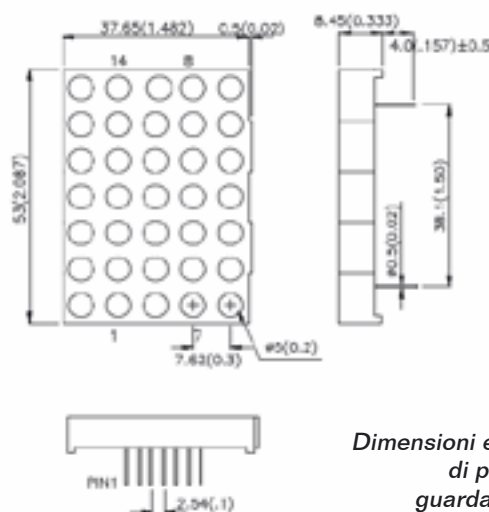
cosiddetti orologi binari, oggi disponibili anche in modelli da polso, che indicano l'ora componendone le cifre mediante led o segmenti di display LCD, nel formato usato dai computer e dai microprocessori (ogni led rappresenta un bit). In pratica, c'è una fila di led per ciascuna cifra e in ognuna di esse i diodi illuminati esprimono il valore, da convertire poi in decimale. Ad esempio, per formare l'ora si usano quattro led e la logica usata prevede che 1 e 0 corrispondano, rispettivamente, ad acceso e spento; come previsto dalla logica binaria, il primo diodo vale 1 (2 elevato alla zero) il secondo 2 (2 elevato alla prima) il terzo 4 (2²) ed il quarto 8 (2³). Dunque, se vediamo accesi i led di peso 1 e 4 vuol dire che sono le cinque. Per i minuti e i secondi, dato che sono entrambi sessanta, occorrono più led o segmenti: sei, per l'esattezza, dato che il primo vale 1, il secondo 2, il terzo 4, il quarto 8, il quinto 16 e il sesto 32; in-

somma, perché con sei bit si può arrivare fino a 64 (2 alla sesta). Così, mantenendo la stessa logica (led acceso = 1, led spento = 0) se vediamo illuminati i led o segmenti 1 e 6, significa che i minuti o secondi sono 33 (1+32); se, invece, ad essere attivi sono i segmenti 5 e 6, vuol dire che i minuti o secondi del caso sono 48 (16+32). L'orologio qui descritto

funziona proprio così: mostra ore, minuti e secondi conteggiati da un apposito modulo RTC interfacciato con un microcontrollore, utilizzando il formato binario; quindi ogni numero è espresso da una certa quantità di led ognuno dei quali vale una potenza di due: 1, 2, 4, 8, ecc. La visualizzazione è nel formato a 24 ore ed ore, minuti e secondi impiegano ognuno due file di led (una per le decine e l'altra per le unità). Ne deriva che la fila più alta è composta da quattro elementi, perché il valore più grande da visualizzare è 9. Per avere un'idea di come avvenga la visualizzazione, immaginate che siano le 20:05:45; la fila di led più a sinistra avrà acceso solamente il secondo led dal basso (2), quella accanto nessun led (0) come quella alla sua destra (0), mentre la quarta fila da sinistra avrà illuminati il primo led e il terzo led dal basso (5). Infine, la quinta fila da sinistra avrà acceso il terzo led dal basso (4) e la sesta il primo e terzo diodo (5).

SCHEMA ELETTRICO

Il nostro orologio si basa su un modulo RTC, che poi è lo stesso da noi presentato nel fascicolo n°



Dimensioni e pinout del display a matrice di punti; i numeri dei pin valgono guardando il componente da dietro.

117; si tratta di un circuito basato su un chip della Maxim-Dallas siglato DS1307 e da un'interfaccia realizzata mediante un piccolo microcontrollore Microchip. Dispone di una linea dati (TX/RX) seriale bidirezionale a livello TTL, di un set di linee per la programmazione in-circuit, di un I²C-bus e dell'ingresso V_{pp}.

Avrete certo capito che l'orologio vero e proprio è l'RTC e il microcontrollore U2 serve a interrogarlo ciclicamente per ottenere le informazioni richieste e a convertire queste ultime nella rappresentazione dell'ora sulla matrice di led. Il micro interroga l'RTC una volta al secondo (tramite l'interfaccia seriale che fa capo alle linee TX ed RX, di tipo bidirezionale; funziona in half-duplex ed è impostata per una velocità di 9.600 bps, 8 bit di dati, nessuna parità e 1 bit di stop), acquisisce e memorizza i dati; trasforma ore, minuti e secondi in valori binari, rappresentati mediante un certo numero di linee di I/O. Per sopperire alla carenza di I/O, implementa il multiplex, al fine di visualizzare in rapida sequenza i dati binari.

Le linee del PIC non pilotano direttamente dei led, perché non sono sufficienti a comandare direttamente, una per punto, i diodi luminosi occorrenti a visualizzare ore, minuti e secondi, dato che servono due led per le decine di ore, quattro per le unità di ore, tre led per decine di minuti e secondi e quattro per unità di minuti e secondi; quindi, almeno 20 diodi ed altrettante linee. All'inizializzazione degli I/O, il firmware del micro U2 imposta RA0, RB0, RB1, RB3, RB4, RC2, RC3 come uscite per il comando delle righe (anodi dei led) della matrice ed RA2, RA3, RA5, RC0 ed RC1 ancora come uscite, ma che saranno dedicate alle colonne

(catodi) della matrice. Infine, B5 e B6 sono inizializzati come ingressi con resistore di pull-up interno: servono a leggere lo stato dei pulsanti di impostazione P1 e P2. Dallo schema si può notare che le righe della matrice vengono pilotate direttamente dalle linee corrispondenti, attive a livello alto, mentre le colonne sono chiuse a zero logico da una serie di line-driver (sette, per l'esattezza) contenuti nell'integrato ULN2803; ogni line-driver è formato da due transistor NPN connessi a Darlington, provvisti di diodo di protezione sul collettore (che in questo caso non serve, dato che non abbiamo a che fare

con carichi induttivi).

Tale struttura circuitale si deve al fatto che pilotando i led della matrice direttamente con gli I/O del microcontrollore, causa la caduta di tensione sulla linea che alimenta (a livello alto) l'anodo e quella che nel passo del multiplex chiude a zero logico il catodo, la differenza di potenziale disponibile sarebbe troppo bassa per far illuminare bene i diodi. Inserendo a valle dei catodi i line-driver, che presentano una resistenza in conduzione nettamente minore di quella degli I/O del micro, si risolve il problema: infatti, malgrado la caduta su linee RA0, RB0, RB1, RB3, RB4,



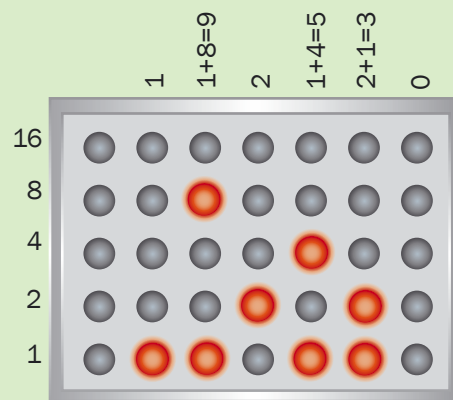
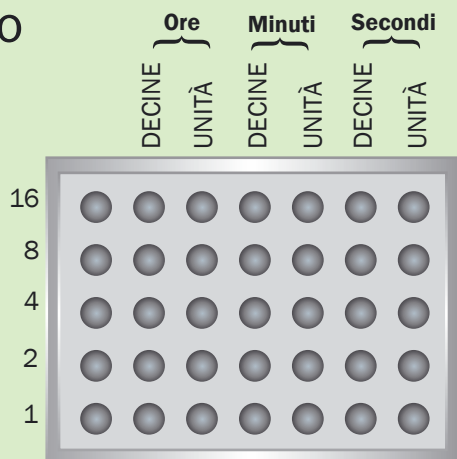
In punta di dito

Siete stanchi dei soliti orologi? Non sopportate l'avere qualcosa al polso, che vi stringe o balla, o che si bagna di sudore e vi dà fastidio per il resto della giornata? Siete allergici ai cinturini metallici? Bene, da oggi potete portare l'orologio in un posto più comodo. No, non vi stiamo proponendo una "cipolla" da estrarre dal panciotto, ma una supernovità tecnologica proposta dalla Timex: TX54 è un orologio da portare in punta di dito; sostanzialmente una finta unghia nella quale si trova un microchip che pilota un sottile display LCD con tanto di illuminatore. In quest'epoca in cui le donne fanno a gara a chi ha le unghie artificiali più suggestive, portare un orologio a forma di unghia è certamente un modo per distinguersi, per vincere quanto a originalità. L'orologio aderisce perfettamente all'unghia del pollice (per la quale è stato pensato) al punto che, spento, risulta praticamente invisibile, perfettamente mimetizzato con il resto della mano; una volta acceso appaiono le scritte illuminate in uno dei diversi colori previsti. Il TX54 ha anche il calendario ed è comandato da un solo pulsante.

Che sorpresa, per chi vi chiedesse l'ora, vedere che invece del polso guardate la punta del dito...

L'ora in binario

Ecco come il nostro orologio mostra l'ora: ogni fila rappresenta, in formato binario (led acceso = 1; led spento = 0) un valore numerico decimale. Per la rappresentazione usiamo due colonne di led per ore, altrettante per i minuti e ancora due per i secondi.



Sono le 19 25' 30"

RC2, RC3 non sia trascurabile, quella dei line-driver, alle correnti in gioco, è trascurabile, quindi la differenza di potenziale sui diodi è tale da garantire una buona luminosità.

L'intero orologio funziona con una tensione continua, di valore compreso tra 9 e 15 volt, applicata ai contatti + e - PWR; un regolatore integrato (si tratta del classico 7805 in contenitore TO-220) provvede a ricavare i 5 volt che servono alla logica (microcontrollore ed RTC).

LE IMPOSTAZIONI

Come tutti gli orologi, anche il nostro prevede i comandi neces-

sari a impostarlo; in questo caso parliamo di P1 e P2. Il primo regola i minuti e il secondo le ore; più esattamente, ogni volta che si preme uno di essi, avanza di un'unità il valore corrispondente. Quindi, ogni pressione di P1 porta avanti i minuti di un'unità, mentre ogni intervento su P2 fa avanzare le ore di unità per volta; durante la fase di modifica, i secondi non avanzano (i led delle due file a sinistra non cambiano). Quando il microcontrollore non rileva più la pressione di alcuno dei tasti per 5 secondi, le modifiche fatte vengono memorizzate nel modulo RTC e divengono effettive; lo notate perché il

display si spegne per circa tre secondi e poi presenta il nuovo orario. L'uscita dalla procedura è evidenziata dalla ripresa dell'avanzamento dei secondi.

REALIZZAZIONE PRATICA

Bene, giunti a questo punto passiamo alle note costruttive. Per realizzare l'orologio binario bisogna per prima cosa preparare il piccolo circuito stampato a doppia faccia sul quale prenderanno posto tutti i componenti; le tracce potete trovarle nel nostro sito Internet www.elettronica.in.it, disponibili sotto forma di file direttamente stampabili in scala 1:1 e quindi più che adatte

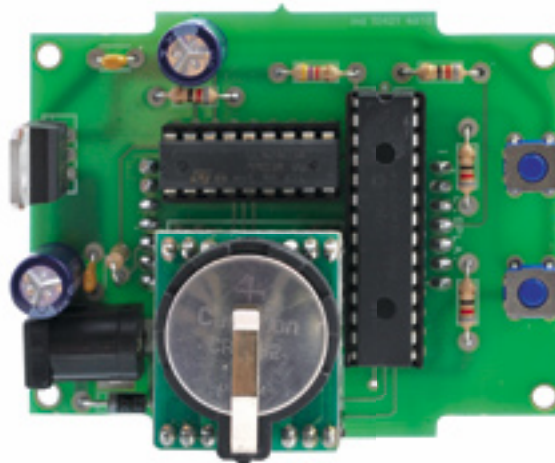
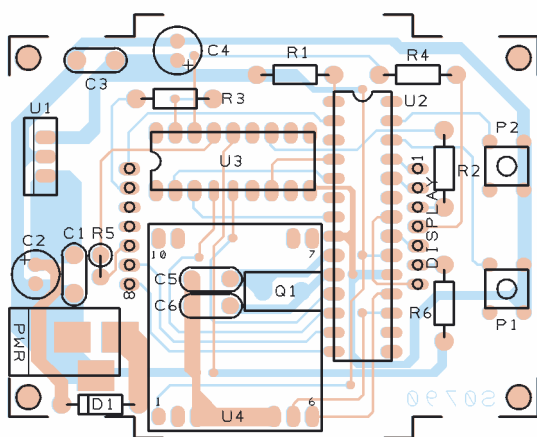


Tecnicismo e tendenza

Strano ma vero, c'è più di un costruttore che realizza orologi binari, persino modelli da polso, dal design avveniristico e accattivante. Certo, non si può dire che siano di immediata leggibilità, però sono senz'altro un modo per distinguersi, per indossare qualcosa che sia insieme utile e originale. Provate a portarne uno al polso e certamente saranno in tanti a chiedervi che cos'è o come funziona. Un consiglio, però: se qualcuno vi domanda che ora è, diteglielo... non fategli vedere l'orologio, del tipo "guarda tu". Se non altro, se volete evitare che vi mandi a quel paese...

In commercio si trova un po' di tutto: orologi con semplici ore, minuti e secondi, altri che hanno sette led in più, cioè uno per ogni giorno della settimana, orologi con datario; poi, i vari costruttori si sono sbizzarriti nel trovare forme di rappresentazione sempre più originali: i valori binari possono apparire non solo sotto forma di led in fila, ma anche di segmenti di varia forma e colori assortiti.

Se cercate un modello da polso, date uno sguardo in Internet a queste pagine Web: <http://www.tokyoflash.com/it/> (sito degli orologi Tokyo Flash) oppure <http://ledwatch.eu> (orologi Logik) o, ancora, <http://www.marea.es> (Marea).



Elenco Componenti:

R1: 4,7 kohm
 R2÷R6: 33 ohm
 C1, C3: 100 nF multistrato
 C2, C4: 470 µF 16 VL elettrolitico
 C5: 15 pF ceramico
 C6: 15 pF ceramico
 D1: 1N4007
 Q1: Quarzo 20 MHz basso profilo

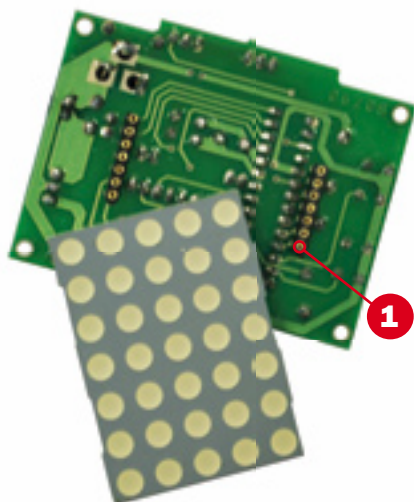
U1: 7805
 U2: PIC16F876A (MF790)
 U3: ULN2803
 U4: Modulo RTC (FT674M)
 P1: Microswitch
 P2: Microswitch
 DISPLAY: Display a matrice 7x5 (TC20-11SRWA)

Varie:
 - Plug alimentazione
 - Zoccolo 9+9
 - Zoccolo 14+14
 - Strip femmina 2 poli (2 pz.)
 - Strip femmina 3 poli (2 pz.)
 - Strip femmina 7 poli (2 pz.)
 - Circuito stampato

alla preparazione delle pellicole per la fotoincisione. Ottenuta la basetta, disponetevi resistenze e condensatori, quindi gli zoccoli per gli integrati; per il modulo RTC dovete prevedere dei connettori femmina single-in-line

a passo 2,54 mm, da due e tre contatti (pezzi di strip femmina). Tutti i componenti prendono posto da un lato, mentre da quello opposto dovete mettere il display a matrice di led, che va montato mediante due strip femmina a passo 2,54 mm, da sette contatti ciascuno; ricordate che il pin 1 è, guardando il display da dietro con a destra il lato riportante due cave, quello più a destra della fila in basso (quello a destra della fila vicina alla sigla stampata sul lato inferiore del corpo). Completato il montaggio, consigliamo di realizzare una mascherina riportante le scritte ore, minuti e secondi, che preveda accanto a ogni led il valore corrispondente (1, 2, 4, 8, 16); va bene anche una lastra di plexiglass trasparente colorata di rosso o fumé. L'ideale è il coperchio (codice BMK153 Futura Elettronica) previsto per il display TC20-11SRWA, che è di colore rosso ed ha le dimensioni adatte ad ospitare l'intero

circuito. Per l'alimentazione potete utilizzare un alimentatore da rete in grado di fornire una tensione, anche non stabilizzata, di valore compreso tra 9 e 15 V e una corrente continua di circa 60 milliampere.



Il display è a matrice di 7x5 led tondi; lo usiamo sdraiato, in modo da visualizzare numeri su sei colonne. Per montarlo usate due strip femmina da 7 contatti l'uno, posti dal lato saldature dello stampato.



per il **MATERIALE**

Il firmware di questo progetto può essere scaricato gratuitamente dal sito della rivista (www.elettronica.in.it). I componenti utilizzati nel circuito costano rispettivamente 18,00 Euro (FT674M), 7,00 Euro (display TC20-11SRWA), 14,00 Euro (MF790, microcontrollore già programmato) e 8,00 Euro (BMK153, mascherina rossa).

Il materiale va richiesto a:
 Futura Elettronica, Via Adige 11,
 21013 Gallarate (VA)
 Tel: 0331-799775 • Fax: 0331-792287
<http://www.futurashop.it>

DAI VOCE

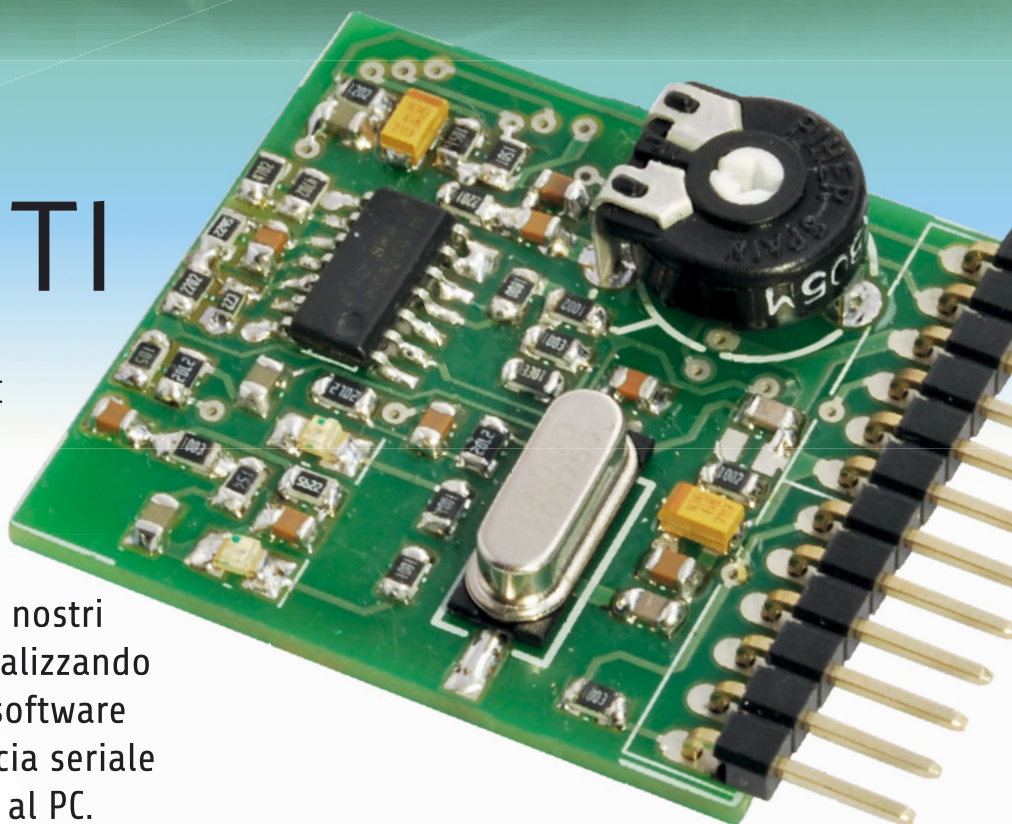
AI TUOI PROGETTI

..... di MARCO LANDONI

Concludiamo la descrizione dei nostri Sintetizzatori Vocali con PIC analizzando il registratore più semplice, il software di gestione ed anche l'interfaccia seriale necessaria per il collegamento al PC.

Nel fascicolo 131 abbiamo introdotto e iniziato a descrivere una nostra nuova idea: realizzare, con l'aiuto di un microcontrollore Microchip, dei moduli per sintesi vocale (*Speech Module*) comandabili da segnali elettrici o pulsanti, esattamente come gli ormai famosi DAST e ChipCorder della ISD e della Winbond. Il perché di questo progetto è presto detto: i vecchi circuiti integrati (che in ogni caso richiedevano un certo numero di componenti esterni) sono sempre più difficili da reperire

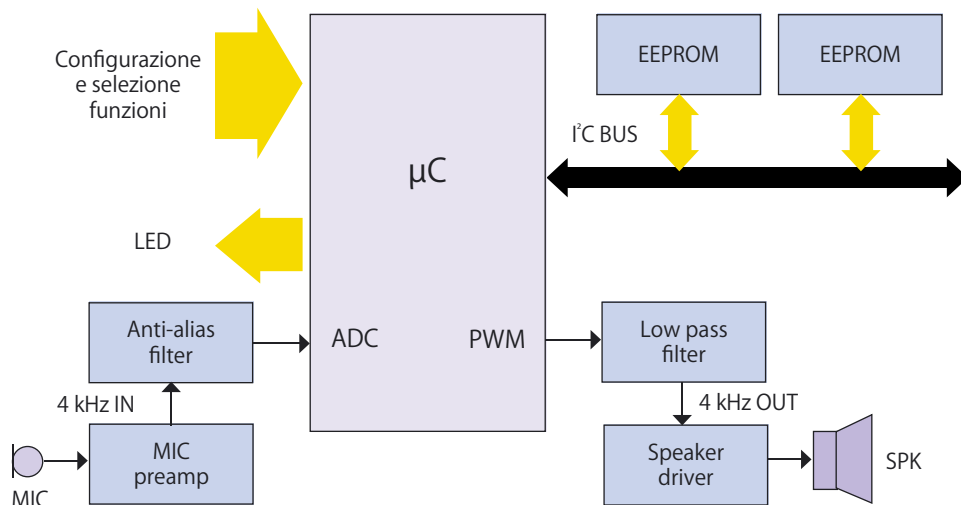
sul mercato; abbiamo dunque pensato di riprodurre le funzioni di questi chip con dei moduli semplici ed economici che sicuramente troveranno un vasto impiego in tutti i dispositivi elettronici "parlanti": il tutto facendo ricorso a componenti comunissimi, microcontrollori Microchip PIC e memorie I²C-Bus. Dopo aver introdotto, nello stesso fascicolo, la filosofia di progetto e la struttura dei dispositivi ed aver descritto dettagliatamente il primo della serie, ossia l'SPC01, nel numero 132, è venuto il momento,



in questa terza e ultima puntata, di occuparci dell'ultimo dei nostri moduli: il registratore/lettore miniatura SPC02. Dapprima ne vedremo lo schema elettrico, poi ne illustreremo le modalità di utilizzo; concluderemo poi parlando della porta seriale, interfaccia comune al modulo SPC01 e la cui spiegazione abbiamo rimandato dalla seconda a questa puntata. Riportiamo per chiarezza d'esposizione lo schema a blocchi dell'SPC02 e le relative caratteristiche; notiamo subito che il modulo è più semplice rispetto al più completo SPC01 già descritto. L'hardware di questo modulo non è personalizzabile come nell'SPC01, perché è dedicato ad un uso specifico e non generico come il precedente; tuttavia la struttura e il funzionamento sono inevitabilmente affini a quelli del "fratello maggiore", poiché

una capacità di registrazione massima di 65 s, campionando ad 8 ksp/s (la frequenza di campionamento dell'SPC02 non è modificabile). Dall'altro canto, però, abbiamo integrato nel modulo un front-end microfonico ed uno speaker driver: a voi non resterà altro da fare che collegare un microfono, uno speaker, tre pulsanti ed alimentare il circuito per cominciare ad utilizzarlo. Tutto questo si può collegare ad un apposito strip di pin a passo 2,54 mm che è stato previsto su un lato del circuito stampato per consentire il montaggio del modulo inserendolo su un altro circuito, quale ad esempio un controllore elettronico per un tornello che debba anche riprodurre un messaggio vocale, un rilevatore di passaggio con avviso vocale, un promobox o un impianto di sicurezza.

Fig.1 - Schema a blocchi del Mini Speech Module SPC02.



il micro, le memorie, gli stadi di filtraggio analogici e la struttura del firmware, sono praticamente identici per i due modelli. In questa configurazione minima troviamo montate solamente due memorie EEPROM, che danno

SCHEMA ELETTRICO

Quanti di voi hanno già avuto modo di leggere la puntata relativa al modulo SPC01, noteranno immediatamente una grande similitudine tra i due circuiti; non si tratta, ovviamente di un

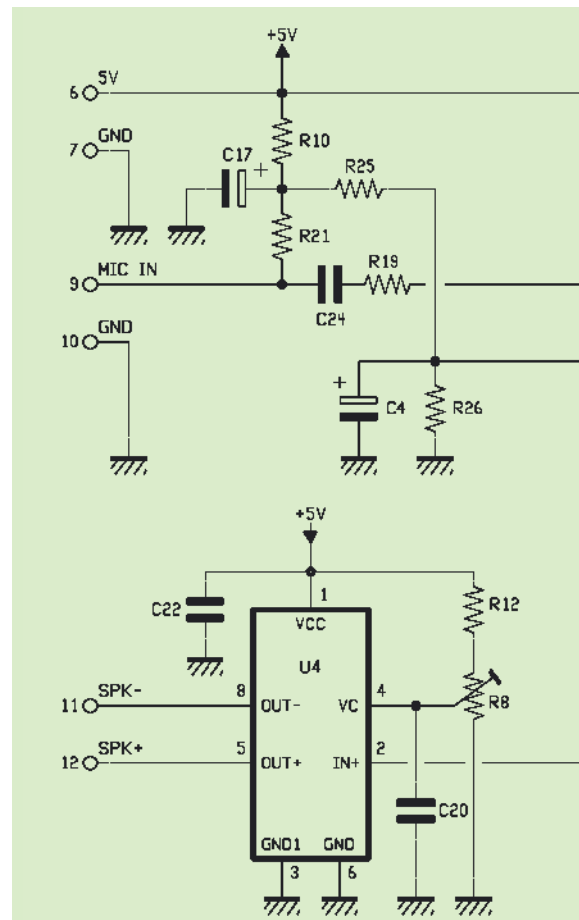
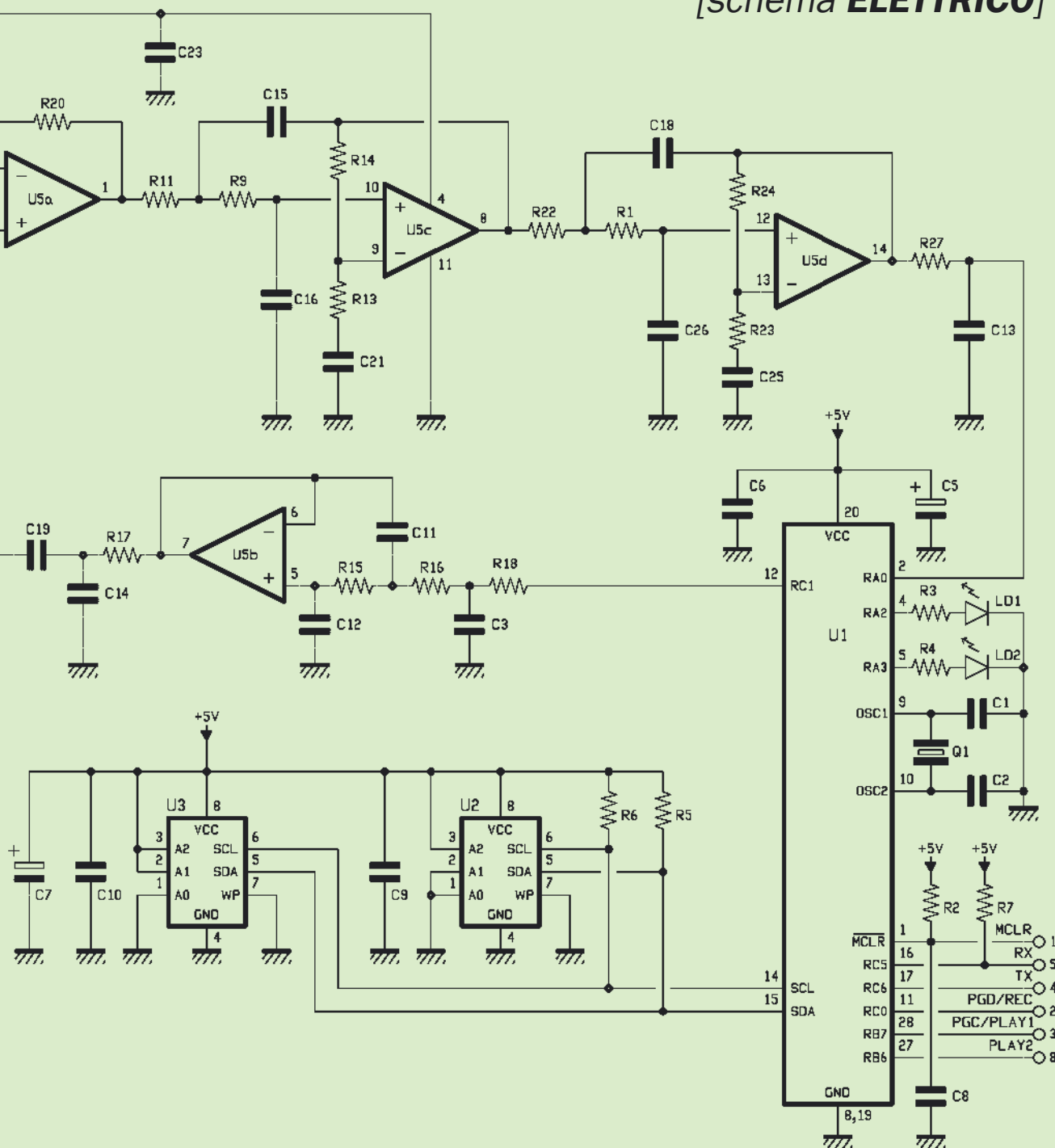


Fig.1 - Il nostro speech processor digitale consta di un microcontrollore Microchip che in registrazione digitalizza il segnale microfonico opportunamente filtrato (per evitare distorsioni) e in lettura converte, mediante l'uscita PWM, i dati campionati in componente audio, amplificata da un finale BF integrato (U4).

caso: i blocchi funzionali sono infatti molto simili tra loro. Il modulo SPC02 è infatti una via di mezzo tra il modulo più complesso SPC01 e la relativa demoboard SPC-DK. Partendo dall'ingresso MIC

[schema **ELETTRICO**]



troviamo un primo amplificatore in configurazione invertente (U5a) che serve ad amplificare di circa 25 dB il segnale captato dal microfono; seguono tale stadio gli altri due operazionali (U5c e U5d) che formano due celle di

Sallen-Key in cascata utilizzate come filtri passa-basso attivi, caratterizzati ciascuno da una frequenza di taglio di 3,1 kHz. Le due celle, unitamente al filtro passa-basso passivo costituito da R27 e C13, formano, esattamente

come per il modulo SPC01, un efficiente filtro passa-basso anti-aliasing del quint'ordine. Il segnale così filtrato viene prelevato ai capi di C13 dall'ingresso del convertitore AD interno al micro, anche in questo caso un



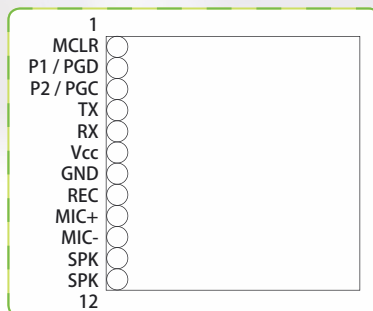
Fig. 2 - Pinout del modulo SPC02 vista dall'alto (lato trimmer e quarzo).

PIC18F2420, il quale si occuperà di campionare il segnale, elaborarlo, salvare i dati in memoria ed effettuare anche tutte le operazioni del procedimento inverso (lettura) ossia estrarre i dati, convertirli in analogico e inviarli all'uscita audio.

Durante la riproduzione di una traccia, il micro utilizzerà la periferica PWM #2 per generare un segnale analogico: sul pin 12 di U1 sarà sempre presente un'onda quadra a frequenza fissa di 195 kHz e, a tale segnale, il firmware varierà il duty-cycle D in modo proporzionale all'ampiezza del segnale vocale decompresso che si deve riprodurre, campione per campione.

Per poter estrarre il segnale modulante (in questo caso la traccia registrata) da un'onda PWM è necessario effettuare una media del segnale stesso, ossia filtrare

mediante una cella passa-basso; infatti esiste una relazione di proporzionalità diretta tra duty-cycle D e valore medio per un segnale ad onda rettangolare. In altre parole, filtrando il segnale si sopprime la portante o, se preferite, si sfrutta il fatto che il condensatore verrà istantaneamente caricato ad un livello di tensione proporzionale alla durata di ogni singolo impulso. Quindi, quando l'onda PWM ha un duty-cycle alto (impulsi lunghi rispetto alle pause) il condensatore assume un livello di tensione più elevato di quando, invece, il duty-cycle



stesso si riduce. Delle operazioni di filtro si occupano le reti che fanno capo ad R18 e C3 e all'operazione U5b, passivo il primo e attivo il secondo. Notate che è indispensabile anteporre al filtro attivo U5b uno stadio passivo per limitare la dinamica di tensione del segnale PWM (che è, elettricamente parlando, un segnale digitale a tutti gli effetti poiché varia tra soli due livelli ossia 0 e Vcc); in caso contrario, si saturerebbe senza dubbio l'ingresso dell'operazione.

Il filtro passivo R17/C14 è utilizzato per rimuovere eventuali residui della frequenza portante

SCHEDA TECNICA

- **Formato di registrazione:** ADPCM 4 bit
- **Durata di registrazione:** 65s @ 8 kbps
- **Frequenza di campionamento:** 8 kbps
- **Risoluzione effettiva:** 8 bit
- **Ingresso analogico:** microfono (40 mVpp_{MAX})
- **Uscita analogica:** Speaker (500 mW @ 16 Ω)
- **Comandi:** PLAY track 1, PLAY track 2, REC
- **Funzioni:** comandi e funzioni via porta seriale (RS232), LED rosso e verde di segnalazione.

dell'onda PWM. Quindi, ai capi di C14, ritroveremo il segnale memorizzato nelle EEPROM decompresso e ricostruito; il condensatore C19 lo preleverà e lo trasferirà all'ingresso dello speaker driver U4, un TDA7052A costruito da NXP Semiconductors, che è in grado di pilotare un piccolo altoparlante da 1W con un'alimentazione singola, senza per questo impiegare grosse ed ingombranti capacità di disaccoppiamento. Inoltre, lo stesso integrato dispone di un controllo di volume esterno: variando la tensione sul pin 4 è possibile variare il guadagno di tensione entro un range di 80 dB. L'interfacciamento con le EEPROM avviene anche in questo caso su bus bifilare SPI e, come è

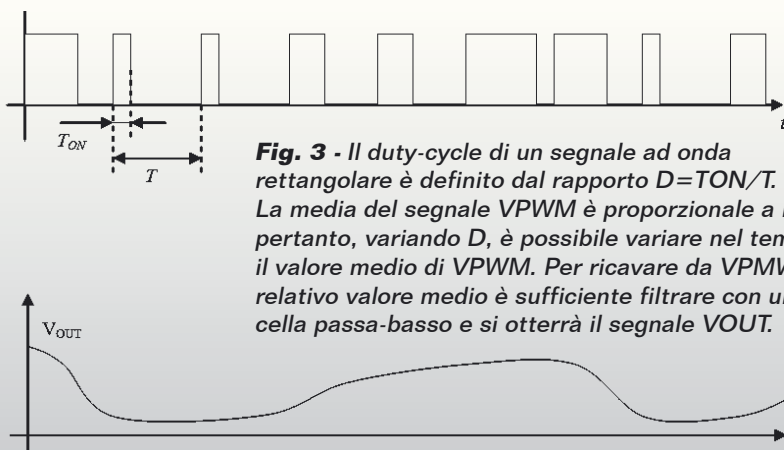


Fig. 3 - Il duty-cycle di un segnale ad onda rettangolare è definito dal rapporto $D=T_{ON}/T$. La media del segnale V_{PWM} è proporzionale a D , pertanto, variando D , è possibile variare nel tempo il valore medio di V_{PWM} . Per ricavare da V_{PWM} il relativo valore medio è sufficiente filtrare con una cella passa-basso e si otterrà il segnale V_{OUT} .

ben visibile dallo schema elettrico, le memorie connesse sono solamente due, ossia U2 e U3. I pulsanti di controllo sono collegati direttamente ai pin del microcontrollore e vengono sfruttati i pull-up interni per forzare lo stato logico di riposo dei pin. Tutto il circuito necessita di un'unica tensione d'alimentazione di 5V stabilizzati.

IL FIRMWARE

Vi parleremo ora, in maniera più ampia di quanto si è fatto fino ad adesso, del firmware implementato nel μC ; descriveremo solamente la sequenza di operazioni effettuate durante la fase di registrazione (poiché quella di riproduzione è molto simile) ed illustreremo, seppure in breve, piccoli stratagemmi che consentono di velocizzare l'esecuzione del programma, poiché in quest'applicazione è fondamentale eseguire il maggior numero di operazioni (non istruzioni!) nel minor tempo possibile. Faremo riferimento a concetti espressi nella prima puntata, quindi potrebbe risultare utile dare un'occhiata all'articolo corrispondente; allo stesso modo potrebbe agevolarvi seguire il codice sorgente.

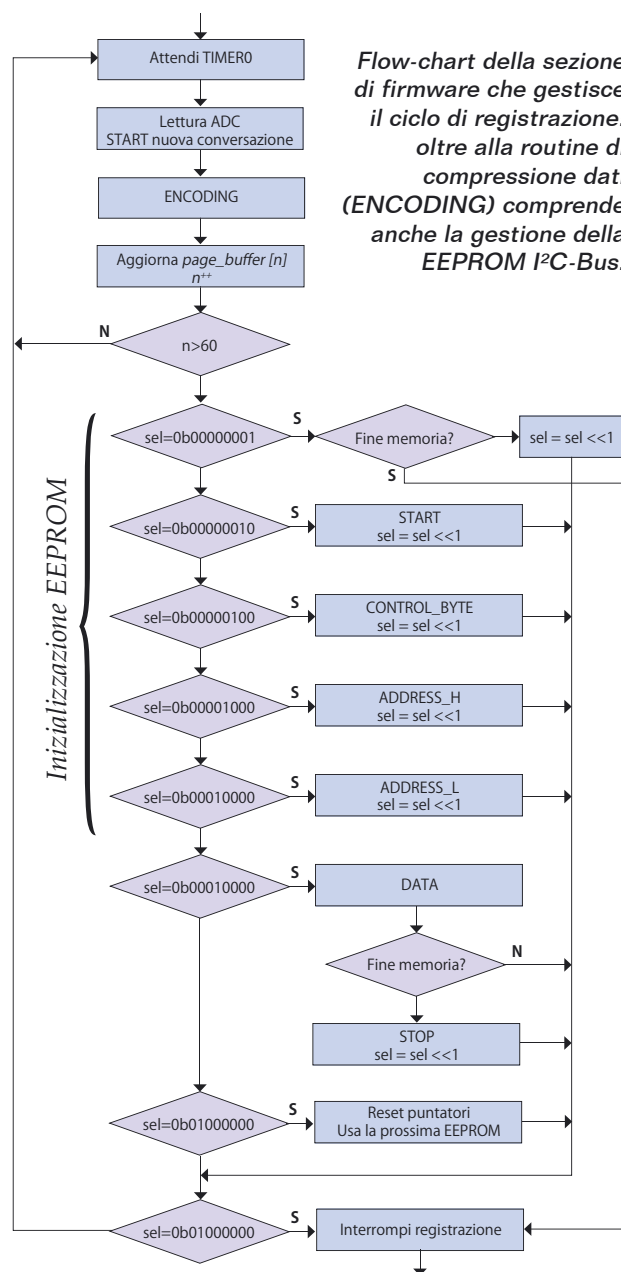
Innanzitutto occorre precisare che si è utilizzato un timer interno al μC (TIMER0) per scandire con precisione la successione delle operazioni: questo è impostato per generare un evento ogni scadere di un intervallo pari a:

$$ts = 1/fs$$

ossia ogni volta che si debba acquisire un campione audio. Si è scelto di non associare al timer una routine di interrupt, ma di andare in polling software sul FLAG di interrupt: questo perché, scrivendo il programma in linguaggio C, ogni volta che fosse stato chiamato l'interrupt sareb-

be stato necessario eseguire parecchie istruzioni inserite in modo automatico dal compilatore per la gestione dello stack, istruzioni che avrebbero sottratto tempo all'esecuzione del programma.

Dopo che è stata verificata la condizione di START di registrazione, il ciclo di operazioni eseguite durante la registrazione è rappresentato dal flow-chart riportato qui a lato. Il ciclo principale viene attivato ad ogni istante di campionamento ts tramite il TIMER0 e, come prima operazione, viene letto il dato convertito dall'ADC. Lo stesso viene fatto ripartire subito dopo per una nuova acquisizione: in tal modo l'ADC potrà effettuare la conversione in background rispetto all'esecuzione del programma, poiché il dato gli sarà richiesto solamente all'inizio del prossimo ciclo. Il campione prelevato dall'ADC viene quindi compresso ed accodato al buffer di memoria interno al μC denominato *page_buffer*; vi ricordiamo che nel *page_buffer* ogni campione è rappresentato su 4 bit, pertanto, nei suoi 128 byte prendono posto 256 campioni vocali. La dimensione di 128 byte per il *page_buffer* non è stata scelta a caso: infatti questa è proprio la dimensione delle pagine di memoria interne alle EEPROM che è possibile scrivere in blocco;



quindi, ogni 256 campioni vocali acquisiti verrà scritta una pagina di EEPROM (di 128 byte). Fintanto che il livello dei dati nel buffer (intesi su 8 bit) è inferiore a 60, cioè poco meno di metà, effettuate le operazioni sopra descritte il programma tornerà all'inizio del ciclo ed aspetterà l'istante esatto per effettuare una nuova conversione; viceversa, se il numero di byte nel buffer è superiore a 60, il programma inizierà a disporre la memoria EEPROM opportuna per riceve-

Gestione EEPROM

Le memorie utilizzate nei registratori SPC01 ed SPC02 sono di tipo seriale con interfaccia I²C-Bus; tale interfaccia utilizza solamente due fili, uno per il clock (SCL) che scandisce la comunicazione, ed uno per i dati veri e propri (SDA); si tratta di un bus sincrono indirizzabile, nel senso che le stesse linee possono essere condivise da più dispositivi, ognuno dei quali è contraddistinto da un proprio indirizzo. Le unità affacciate al bus possono essere master o slave: il master è quello che decide quando iniziare una

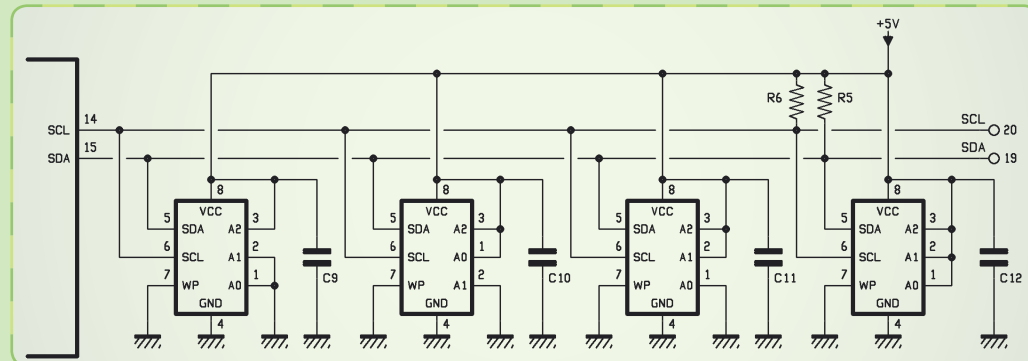


Fig. A- Il bus I²C utilizzato nel registratore SPC01.

ze dovranno essere piccole, altrimenti, causa le capacità parassite delle strutture MOS che realizzano i chip, rallenteranno la commutazione. Il bus viaggia nominalmente fino ad un massimo di 400 kHz

funzioni sono praticamente trasparenti per il programmatore e si riducono a dei semplici set o reset di specifici FLAGS. Vediamo ora come avviene la lettura/scrittura di celle o banchi di memoria nel progetto in questione (vi consigliamo di tenere sott'occhio i data sheet del PIC18F2420 e della memoria 24XX1025). Innanzi tutto occorre inizializzare la porta MSSP (Master Synchronous Serial Port) del PIC in modo

adeguato. La prima istruzione setta l'uso dei pin specifici, abilita la periferica e la configura come master; la seconda istruzione mette la porta in condizione di IDLE mentre la terza setta la frequenza di lavoro del bus. A questo punto la porta è operativa. Per poter andare a scrivere un dato su una delle EEPROM presenti sul bus dovremo eseguire delle operazioni di lettura/scrittura dati sul bus

Listato 1

```
// ----- I2C bus -----+
SSPCON1 = 0b00101000; // set MASTER, enable I2C
SSPCON2 = 0b00000000; // idle...
SSPAD = 0x18; // fbus = 400kHz
```

sessione di comunicazione ed è normalmente uno solo; lo slave si limita a rispondere alle richieste del master, ovvero esegue i comandi e riceve o trasmette i dati richiesti. A livello elettrico tutti i dispositivi sono connessi con porte bidirezionali che, quando configurate come uscite, devono essere del tipo open-drain (pertanto sono necessarie delle resistenze di pull-up comuni a tutto il bus). Tanto più sarà elevata la velocità del bus, tanto più queste resisten-

(velocità del clock SCL) anche se, sotto particolari condizioni, è possibile raggiungere anche 1 MHz. Giacché la linea dati è unica, è possibile far transitare un dato per volta ed in modo sincrono con il segnale di clock. Inoltre il protocollo prevede l'invio di segnali di acknowledge (ACK) e delle particolari temporizzazioni che segnalano l'inizio di una trasmissione, il proseguimento e la fine. Fortunatamente, utilizzando gli attuali microcontrollori, parecchie

Listato 2

```
// START -----
SSPCON2bits.SEN = ENABLE; // START bit
while(SSPCON2bits.SEN); // wait slave...

// set write mode -----
SSPCON2bits.ACKDT = ZERO; // rispondi con ACK
PIR1bits.SSPIF = ZERO;
SSPBUF = 0xA0 | (M << 1); // control byte,
// M = memoria da indirizzare
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

// set address to write -----
PIR1bits.SSPIF = ZERO;
SSPBUF = ADDR_H; // b15 - b8
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

PIR1bits.SSPIF = ZERO;
SSPBUF = ADDR_L; // b7 - b0
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

// write data -----
PIR1bits.SSPIF = ZERO;
SSPBUF = DATA; // scrivi la cella
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

// STOP -----
SSPCON2bits.PEN = ENABLE; // STOP bit
while(SSPCON2bits.PEN); // wait slave...
```

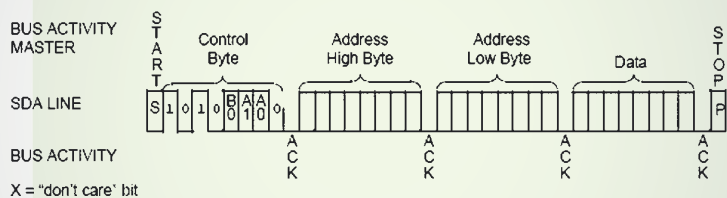


Fig. B - Comandi e dati da inviare per effettuare una scrittura in memoria. La scrittura vera e propria inizia nel momento in cui si presenta il segnale di STOP.

re un banco di dati. Poiché non sarebbe possibile trasferire alla memoria 128 byte in maniera consecutiva senza interferire temporalmente con l'operazione

di campionamento, il trasferimento dati è stato "splittato" in tanti "step" gestiti da una piccola macchina a stati: in altre parole, ad ogni ciclo della routine di

registrazione viene effettuato il trasferimento di un singolo dato (di 8 bit). Non va però dimenticato che per poter scrivere nella EEPROM è necessario asserire

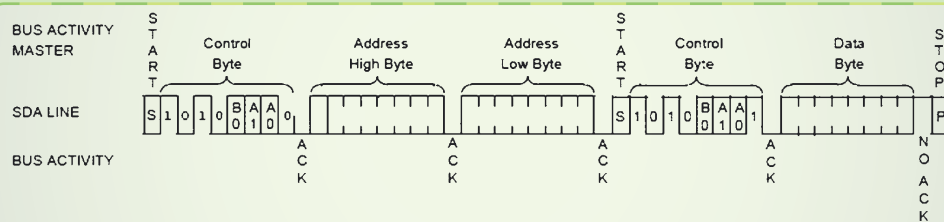


Fig. C - Comandi e dati da inviare per effettuare una lettura della memoria.

secondo quanto indicato nel data-sheet della memoria. In particolare, volendo scrivere un singolo byte in una specifica locazione di memoria, è necessario:

- inviare una segnalazione di START;
- inviare un CONTROL BYTE nel quale è specificato l'integrato a cui si vuole accedere;
- inviare due byte di indirizzo;
- inviare il byte da scrivere;
- inviare una segnalazione di STOP.

Il control byte (c8 ÷ c0) è strutturato nel seguente modo:

i primi 4 bit c7 ÷ c4 sono sempre fissi e sono costituiti da una sequenza fissa di 1 e 0 alternati, il bit c3 è il 17° bit di indirizzamento che, assieme ai sedici bit specificati nei due byte indirizzi, permette di indirizzare 217 celle per un totale di 128 kbyte (1 Mbit) ossia tutta la memoria.

I due bit c2 e c1 specificano l'indirizzo assegnato alla EEPROM tramite i pin A0 e A1 (il pin A2 va sempre mantenuto a Vdd); infine, il bit c0 se resettato segnala alla memoria la modalità lettura, viceversa indicherà la modalità di scrittura.

Il codice che realizza tutto questo si riduce a poche righe grazie all'integrazione tecnologica effettuata nel microcontrollore (vedi Listato 2). Dall'istante in cui viene prodotto il segnale di STOP, è necessario attendere almeno 5 ms per lasciare alla memoria il tempo di salvare il

dato prima di poter accedere allo stesso integrato; è invece possibile accedere immediatamente ad uno qualsiasi degli altri integrati presenti sul bus. La lettura di una cella di memoria è leggermente più complicata poiché è necessario specificare due control byte per ogni cella letta: il primo serve per scrivere nel buffer della memoria l'indirizzo che si vuole leggere, mentre il secondo è necessario per far entrare la memoria in modalità "lettura".

Anche in questa sequenza di comandi su bus I²C è necessario specificare le condizioni di START, STOP e ACK. La condizione contrassegnata come START prima del secondo control byte in Fig. C è in realtà una condizione di RESTART, poiché interviene tra uno START dato in precedenza ed uno STOP finale.

È possibile leggere o scrivere più dati in sequenza senza dover accedere ogni volta ad ogni singola cella; per la scrittura si procede esattamente come esposto, con una differenza: il blocco di codice write data andrà ripetuto tante volte quante sono le celle da scrivere.

La scrittura dovrà terminare comunque sempre inviando il comando di STOP. Facciamo presente che è possibile scrivere in sequenza a patto che non si sbordi dalla singola pagina di memoria: partendo dall'indirizzo 0x0000 la prima pagina finirà all'indirizzo 0x007F, la seconda inizierà

dal l'indirizzo 0x0080 e terminerà all'indirizzo 0x00FF e così via a passi di 128 byte per tutto lo spazio d'indirizzamento. Anche in lettura è possibile leggere in sequenza;

analogamente a prima sarà necessario ripetere quanto racchiuso nel blocco read data tante volte quante sono le celle da leggere. L'unica accortezza da adottare è che ad ogni byte letto va asserito un ACK, mentre al termine della lettura dell'ultimo byte va previsto un NACK (come nell'esempio sopra riportato). La lettura contigua è possibile dall'indirizzo 0x00000 all'indirizzo 0x0FFFF senza interruzioni e da 0x10000 a 0x1FFFF.

Listato 3

```
// START -----
SSPCON2bits.SEN = ENABLE; // START bit
while(SSPCON2bits.SEN); // wait slave...

// set write mode -----
SSPCON2bits.ACKDT = ZERO; // rispondi con ACK
PIR1bits.SSPIF = ZERO;
SSPBUF = 0xA0 | (M << 1); // control byte,
// M = memoria da indirizzare
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

// set address to start read -----
PIR1bits.SSPIF = ZERO;
SSPBUF = ADR_H; // b15 - b8
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

PIR1bits.SSPIF = ZERO;
SSPBUF = ADR_L; // b7 - b0
while(!PIR1bits.SSPIF); //
while(SSPCON2bits.ACKSTAT); // wait ACK

// set read mode -----
SSPCON2bits.RSEN = ENABLE; // RE-START bit
while(SSPCON2bits.RSEN); // wait start asserted
PIR1bits.SSPIF = ZERO;

SSPBUF = 0xA1 | (M << 1); // re-assert control byte
// in read mode
while(!PIR1bits.SSPIF);
while(SSPCON2bits.ACKSTAT); // wait ACK

SSPCON2bits.ACKDT = ONE; // rispondi con NO-ACK

// read at address -----
PIR1bits.SSPIF = ZERO;
SSPCON2bits.RCEN = ENABLE; // enable read
SSPSTATbits.BF = ZERO;
while(SSPCON2bits.RCEN); // aspetta completamento lettura

DATA = SSPBUF; // leggi dato

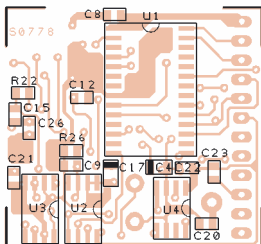
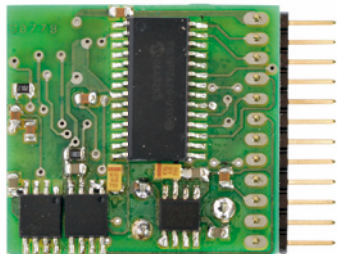
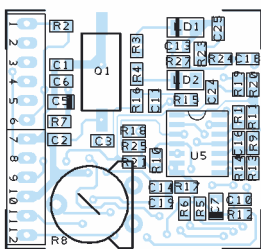
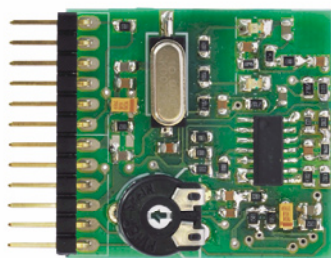
SSPCON2bits.ACKEN = ENABLE; // assert NACK
while(SSPCON2bits.ACKEN); // wait...

// STOP -----
SSPCON2bits.PEN = ENABLE; // STOP bit
while(SSPCON2bits.PEN);
SSPCON2bits.ACKDT = ZERO; // rispondi con ACK
```

prima alcuni byte di controllo e specificare l'indirizzo cui accedere (vedere *Gestione EEPROM*, in queste pagine), pertanto anche queste operazioni sono

state tra loro separate ed eseguite una per volta ad ogni ciclo della routine principale. In pratica, dopo che nel *page_buffer* è stato scritto il

sessantesimo byte, il programma inizierà a dialogare anche con la EEPROM, inviandole, ad ogni ciclo della routine di registrazio-



Elenco Componenti:

R1: 27 kohm 5% (0805)	C6: 100 nF ceramico (0805)
R2: 100 kohm 5% (0805)	C7: 10 μ F 6,3 VL tantalio
R3, R4: 1 kohm 5% (0805)	C8: 1 nF ceramico (0805)
R5, R6: 1,5 kohm 5% (0805)	C9: 100 nF ceramico (0805)
R7: 10 kohm 5% (0805)	C10: 100 nF ceramico (0805)
R8: Trimmer MO 10 kohm	C11, C12: 2,2 nF ceramico (0805)
R9: 56 kohm 5% (0805)	C13, C14: 1 nF ceramico (0805)
R10: 150 ohm 5% (0805)	C15: 2,2 nF ceramico (0805)
R11: 22 kohm 5% (0805)	C16: 1 nF ceramico (0805)
R12: 12 kohm 5% (0805)	C17: 47 μ F 6,3 VL tantalio
R13, R14: 47 kohm 5% (0805)	C18: 2,2 nF ceramico (0805)
R15, R16: 27 kohm 5% (0805)	C19÷C25: 100 nF ceramico (0805)
R17: 2,2 kohm 5% (0805)	C26: 2,2 nF ceramico (0805)
R18: 3,3 kohm 5% (0805)	LD1: led rosso (0805)
R19: 27 kohm 5% (0805)	LD2: led verde (0805)
R20: 1 Mohm 5% (0805)	U1: PIC18F2420 (SOIC28)
R21: 10 kohm 5% (0805)	U2, U3: 24LC1025-I/SM (SOIC8)
R22: 27 kohm 5% (0805)	U4: TDA7052A (SOIC8)
R23: 150 kohm 5% (0805)	U5: MCP6002 (SOIC8)
R24÷R26: 100 kohm 5% (0805)	Q1: Quarzo 10 MHz (HC49/4H)
R27: 56 kohm 5% (0805)	
C1, C2: 15 pF ceramico (0805)	Varie:
C3: 4,7 nF ceramico (0805)	- Strip maschio 12 poli
C4: 22 μ F 6,3 VL tantalio	- circuito stampato
C5: 10 μ F 6,3 VL tantalio	- circuito stampato

ne, un byte oppure un comando. Così facendo, quando si inizierà a riempire la seconda metà del *page_buffer*, contemporaneamente si inizierà ad inviare il primo byte dello stesso buffer verso la EEPROM. A tal punto, per

completare il buffer, resteranno da acquisire 128 campioni audio (64 byte compressi) e verso la memoria dovranno essere trasferiti proprio 128 byte. Ora, trasferendo un byte ad ogni ciclo di acquisizione AD otterremo

che l'ultimo campione compresso verrà aggiornato appena prima di essere trasferito alla memoria (ciò accadrà nello stesso ciclo della routine di registrazione); in pratica, l'acquisizione ed il posizionamento dei 256 campioni vocali nel *page_buffer* termina in corrispondenza del trasferimento verso la EEPROM del 128° byte dello stesso buffer.

A questo punto verrà chiusa la comunicazione con la memoria, la quale registrerà la pagina appena trasferita; il prossimo ciclo di routine principale ricomincerà a riempire il buffer partendo dall'inizio. Tale procedura continuerà fintanto che non verrà premuto il pulsante STOP o il μ C si accorgerà che la memoria disponibile è terminata. Qualcuno, osservando il codice sorgente, potrebbe domandarsi perché la struttura della macchina a stati non è stata realizzata utilizzando un costruito *case* ed incrementando una variabile, come sarebbe più naturale fare, ma, piuttosto, utilizzando una serie di *if* ed una variabile inizializzata a 0x01 che ad ogni passo viene "shiftata" di un bit. Ebbene, il motivo è che un test del tipo *if(sel & 0b00000100)* è compilato con una sola istruzione di test sul bit *b2* della variabile *sel* (1 istruzione), viceversa, per effettuare la stessa operazione un costruito *case* richiede almeno 5 o 6 istruzioni macchina. Il fatto di shiftare la variabile *sel* anziché incrementarla, permette di effettuare sempre e solo test su singolo bit; il prezzo da pagare è che non si può spezzare il flusso del programma in più di otto vie.

REALIZZAZIONE PRATICA

Per contenere le dimensioni del modulo abbiamo optato per un montaggio SMD, scegliendo componenti di taglio 0805, in

Dopo vinile e CD, ecco l'album su SD

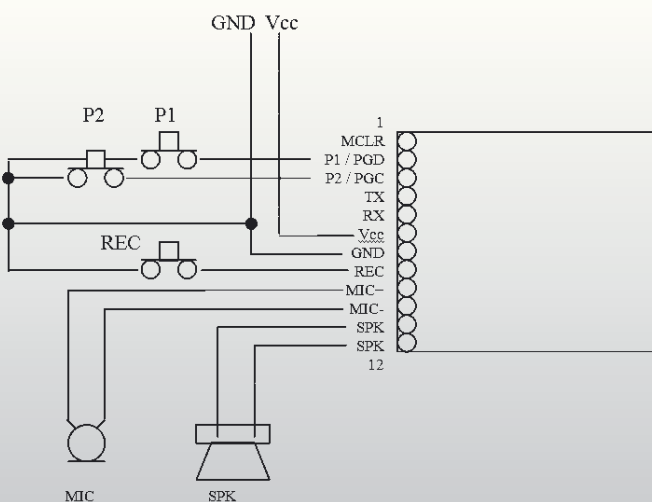
modo da limitare le difficoltà che incontrerete nel montaggio. Vi consigliamo, una volta in possesso del circuito stampato, di cominciare saldando il microcontrollore U1, già programmato; utilizzate al solito una buona dose di flussante e possibilmente dello stagno da 0,25 mm. Procedete quindi montando tutti gli altri integrati: le memorie, i due operazionali e l'amplificatore audio per l'altoparlante. Montate quindi tutti i componenti passivi, facendo attenzione ad inserirli al posto corretto, poi montate i condensatori al tantalio, il diodo luminoso e, per ultimo, il quarzo SMD da 10 MHz. Inserite quindi anche il trimmer per la regolazione del volume e, nel caso vogliate utilizzarli, saldate i pin strip alle relative piazzole sullo stampato. Il circuito a questo punto è pronto e, non necessitando di taratura, se non avete commesso errori sarà anche già funzionante; dovrete solamente collegare ai relativi pin un microfono, un altoparlante, i pulsanti e alimentare il tutto come schematizzato in Fig. 4. I pin 4 (TX) e 5 (RX) andranno utilizzati nel caso vogliate sfrut-

Sono passati 25 anni da quando, con meraviglia e stupore, abbiamo ascoltato per la prima volta la musica uscire da quei dischetti lucenti che un po' ci portavano ai film di fantascienza; eppure l'era del compact disc sembra stia per finire. Infatti SanDisk, leader nel settore delle SD-Card ha annunciato la distribuzione delle Slot Music, album musicali in formato mp3 memorizzati in memory card di tipo microSD (quelle utilizzate anche nei cellulari) che saranno in vendita negli Stati Uniti forse già a Natale, anche se c'è un precedente europeo: l'album e2 (Eros al quadrato) di Eros Ramazzotti è disponibile come SD Card da 1 giga precaricata su cellulari abilitati. La qualità sarà migliore di quella di un CD, dato che è prevista una codifica a 320 kb/s contro i 44,1 kb/s del CD, che con il quadruplo sovracampionamento (four time oversampling) arrivano a 160. L'idea di proporre gli album musicali su memory card nasce da una considerazione: ormai moltissime persone usano i lettori mp3 per ascoltare musica in cuffia, perché permettono di beneficiare dell'elevata capienza dei supporti digitali, che consente di ascoltare ore di musica con dispositivi di dimensioni paragonabili a quelle di un accendino: infatti in mp3 un minuto di brano consuma



circa 1 MB di memoria. Per ascoltare la musica su un mp3, chi compera un disco deve leggerlo, convertirlo in mp3 e scaricare i file corrispondenti nella memoria del lettore o su una memory card; qualcuno ha dunque pensato di far risparmiare questa fatica e, proprio per questo è previsto che le Slot Music siano prive di Drm, i lucchetti digitali che impediscono la copia delle canzoni scaricate da alcuni store come iTunes. Infatti oggi come oggi hanno poco senso, perché esistono fior di programmi per copiare anche i dischi protetti. Al momento le Slot Music sono un'alternativa ai CD: verranno offerte da quattro grandi produttori come Sony, Warner, Emi e Universal. Grazie a piccoli adattatori USB, potranno essere lette anche su un normale PC o su lettori che dispongono della sola USB. Il prezzo di vendita negli USA è stimato sia tra i 7 e i 10 dollari; la capacità sarà di almeno 2 GB, considerato che per un album della durata di un'ora serve una capacità di almeno 1,2 Gb.

Fig. 4 - Schema di collegamento del modulo SPC02: oltre all'alimentazione stabilizzata di 5V, occorre collegare una coppia di pulsanti P1 e P2 per il PLAY delle tracce, un pulsante per la registrazione REC, un microfono e un piccolo altoparlante.



tare le funzioni controllate da PC per la gestione della memoria del registratore (vedi il relativo paragrafo).

UTILIZZO DEL REGISTRATORE

Come già accennato, questo modello di registratore non dispone volutamente di tutte le funzioni del modulo più grande; quindi non potremo variare la frequenza di campionamento, ripetere le tracce, saltare da una all'altra e così via. Al contrario, avremo la possibilità di registrare solamente due messaggi di durata qualsiasi.

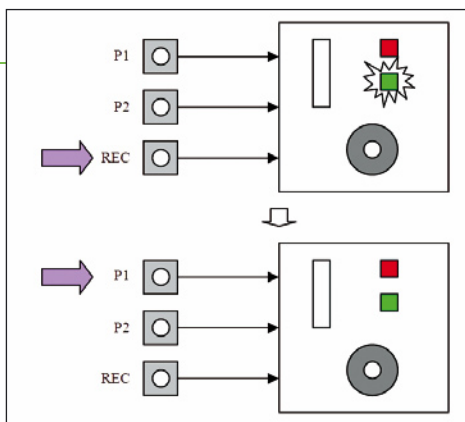


Fig. 5 - Alla pressione del pulsante REC si entrerà in modalità registrazione ma la memorizzazione del messaggio avverrà solamente alla pressione di P1 o P2.

Per registrare il primo messaggio è sufficiente premere il pulsante REC: il led rosso si accenderà segnalando che è in corso una registrazione e, contemporaneamente, lampeggerà il led verde, indicando che il registratore è in attesa del segnale d'inizio registrazione. Tale segnale viene dato premendo uno dei pulsanti P1 o P2: da quell'istante, tutto ciò che verrà captato dal microfono sarà memorizzato. Consigliamo di parlare ad una distanza compresa tra 50 cm e 1m dal microfono, con voce ferma. Per terminare la registrazione dovrete premere nuovamente REC; allora i led si spegneranno. I pulsanti P1 e P2,

Fig. 6 - La riproduzione avviene semplicemente premendo il pulsante relativo alla traccia da riprodurre (P1 per la prima, P2 per la seconda). Gli stessi pulsanti sono utilizzati per mettere la riproduzione in pausa.

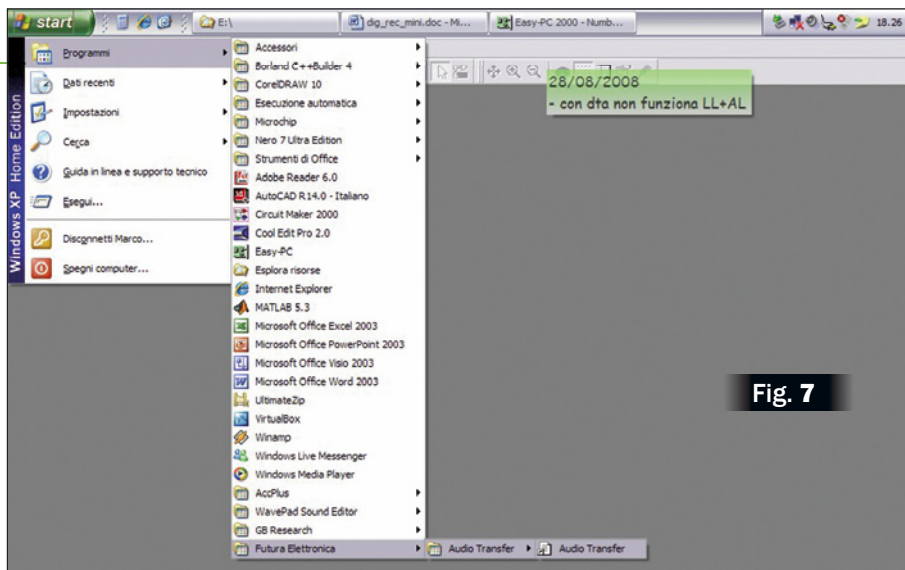
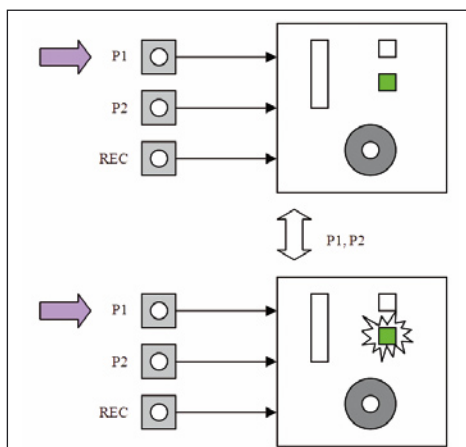


Fig. 7

durante la registrazione, servono anche per inserire una pausa ossia, premendone uno, la registrazione verrà sospesa e riprenderà ad una nuova pressione di uno degli stessi. A questo punto, se premerete il pulsante P1 potrete ascoltare quanto registrato nella posizione 1: la riproduzione è indicata dall'accensione del solo led verde. Durante la riproduzione una pressione di un o dei due pulsanti di PLAY (P1 o P2) metterà il registratore in pausa; per riprendere la traccia sarà sufficiente ripremere uno tra P1 o P2. Per registrare anche la seconda traccia, sarà sufficiente ripetere le stesse operazioni appena descritte per la registrazione della prima: sarà il micro che si occuperà di accodare la nuova traccia a quella già esistente. Nel caso in cui abbiate registrato entrambi i brani disponibili e tentiate di registrarne un altro, il registratore automaticamente cancellerà tutta la memoria e inizierà la registrazione di una nuova traccia #1. Il potenziometro R8 vi servirà per regolare il volume durante la riproduzione.

FUNZIONI GESTITE DAL COMPUTER

In questa sezione descriveremo come utilizzare il programma per PC espressamente realizzato per leggere e scrivere nella memoria dei due moduli registratori

SPC01 ed SPC02, con la premessa che tutte quante le procedure che andremo a descrivere sono identiche per entrambi i moduli. Il programma per PC consente, tramite porta seriale, di leggere e scrivere il contenuto dell'intera memoria del registratore: questa funzione risulta utile ad esempio per creare una sorta di libreria di frasi per il vostro modulo registratore, nella quale potrete scegliere di volta in volta quale utilizzare.

Un'altra possibilità è quella di creare una serie di moduli che riproducano tutti la stessa frase, senza dover ripetere per ogni registratore le stesse frasi. In questa evenienza, una volta che avrete registrato tutto quanto vi occorre su uno dei moduli, potrete leggere e salvare il contenuto di tutta la memoria in un file su PC; fatto ciò potrete copiare l'intero file nella memoria ancora non programmata di tutti gli altri moduli che vorrete, per così dire, clonare. Ma vediamo subito come procedere: dopo aver alimentato e connesso il modulo al PC tramite una porta seriale, la prima cosa da fare sarà quella di installare il software che potrete scaricare dal nostro sito. La procedura d'installazione automatica vi guiderà passo-passo. Al termine dell'installazione potrete cercare il programma *Audio Transfer* dal menu *Programmi* di Windows e

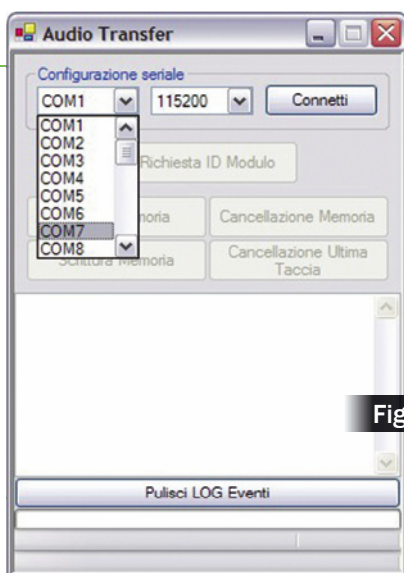


Fig. 8

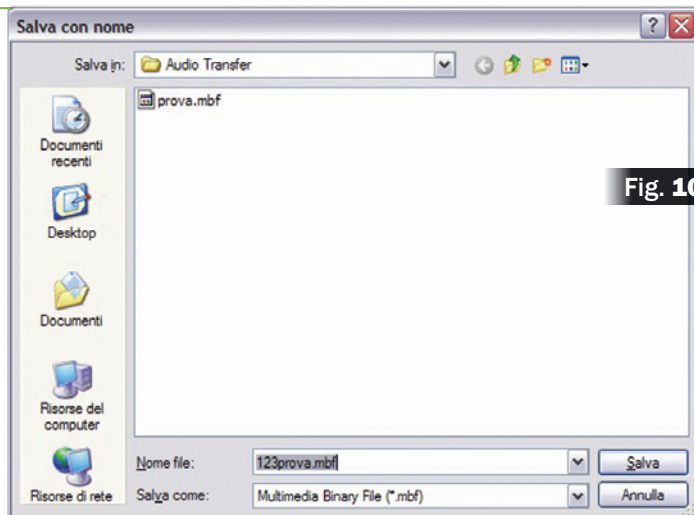


Fig. 10

clicare sul suo nome o icona (Fig. 7). Immediatamente vi comparirà una finestra come quella illustrata nella Fig. 8.

Selezionate quindi la porta COM alla quale avete connesso il registratore ed impostate la velocità di comunicazione a 115,2 kbaud; cliccate, infine, sul bottone "connetti". Se la porta selezionata è corretta, verranno sbloccati tutti gli altri pulsanti, altrimenti un messaggio di errore vi segnalerà di provare con un'altra porta.

Nel caso vogliate uscire dal programma, vi consigliamo sempre di disconnettere prima il dispositivo tramite l'apposito pulsante. Dopo che avrete aperto la porta COM, potrete fare clic sul pulsante *Richiesta ID Modulo*: tale comando vi consentirà di sapere che tipo di modulo avete collegato al PC (se SPC01 o SPC02) e il numero di memorie installate nel modulo stesso (Fig. 9). Lo stesso comando serve allo stesso tempo anche per controllare che la comunicazione seriale funzioni correttamente.

Tramite i due pulsanti posti sulla destra della finestra di dialogo, potrete scegliere se cancellare completamente la memoria del registratore oppure solamente l'ultima traccia registrata. In entrambi i casi un messaggio vi avviserà della corretta esecuzione dell'operazione; durante l'esecuzione del comando i led sulla



Fig. 9

scheda lampeggeranno contemporaneamente. Tenete presente che, una volta che cancellerete una o più tracce dalla memoria, queste non potranno più essere recuperate. Infine, impartendo ripetutamente il comando *Cancellazione ultima traccia*, potrete cancellare anche in questo modo tutta la memoria, procedendo a ritroso. Il pulsante *Letture memoria* consentirà di leggere tutto il contenuto delle memorie montate a bordo del registratore; premendolo comparirà una finestra di dialogo che vi permetterà di salvare in una cartella e con un nome file a scelta il contenuto della memoria (Fig.10).

Una volta assegnato un nome al

file, fate clic sul pulsante *Salva*: immediatamente inizierà il trasferimento dei dati verso il PC. Durante il trasferimento (che durerà grosso-modo una quindicina di secondi per ogni memoria installata) vedrete il led verde sempre acceso, mentre quello rosso lampeggerà ad ogni blocco di dati inviato. Sull'interfaccia grafica, una barra d'avanzamento vi indicherà visivamente la percentuale di trasmissione completata (Fig. 11).

Al termine dell'operazione verrà visualizzato il messaggio "Letture OK". La scrittura in memoria di un file precedentemente salvato si effettua in modo analogo:



Fig. 11

L'interfaccia seriale

Per poter utilizzare le funzioni di controllo da PC dei registratori SPC01 ed SPC02, è necessario che utilizziate la porta seriale. Dato che i pin RX e TX della seriale del modulo non sono direttamente

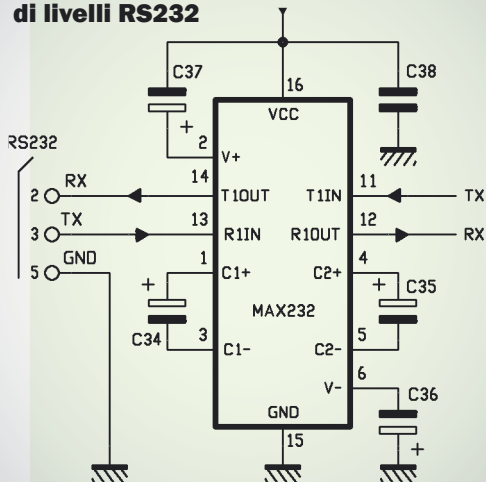
utilizzare una porta seriale RS232 oppure una USB: nel primo caso dovrete impiegare un semplice convertitore di livello come il classico MAX232, mentre volendo utilizzare una porta USB

sarà necessario dotarsi di una scheda d'interfaccia un po' più complessa basata, per esempio, sul chip FTDI FT232R. Gli schemi elettrici li potete trovare nelle figure qui di seguito; tenete presente che i pin contrassegnati con RX e TX andranno collegati direttamente con

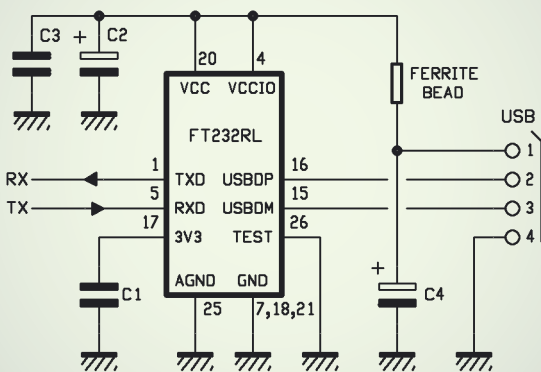
compatibili con i segnali delle porte standard del PC (sono a livello TTL, mentre il computer dialoga a livello RS232) dovete impiegare un'apposita unità di conversione. Potrete scegliere se

le controparti presenti sulle schede SPC01 o SPC02. Allo stesso modo andranno collegati anche i fili per l'alimentazione, che potrete prelevare dalla scheda registratore.

Convertitore di livelli RS232



Convertitore seriale - USB



in primo luogo dovrete fare clic sul solito pulsante *Scrittura memoria* ed immediatamente vi comparirà un'altra

finestra di selezione, dove dovrete scegliere uno tra i file .mbf che avrete salvato in precedenza (Fig. 12). A questo

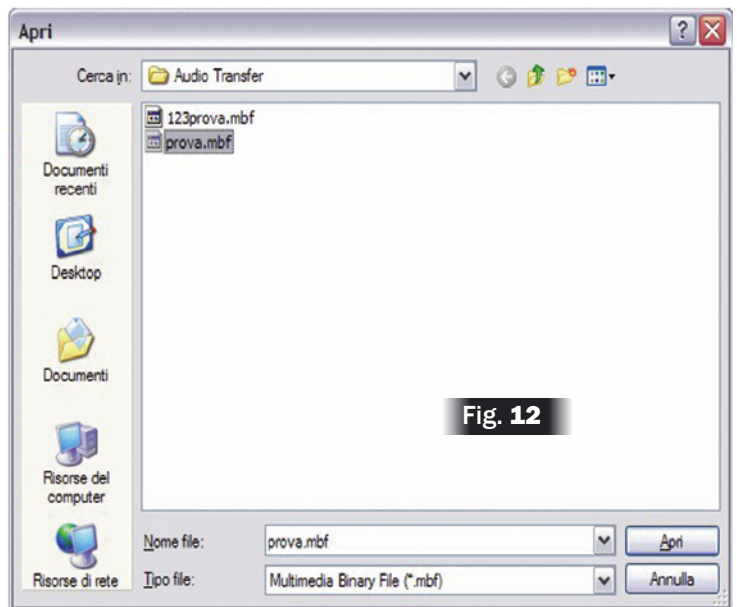
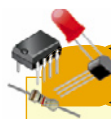


Fig. 12

punto, facendo clic col mouse sul pulsante *Apri*, inizierà il download del file nella memoria del registratore: anche in questo caso una barra di stato indicherà il progresso dell'operazione, esattamente come per la lettura e, contemporaneamente, sul registratore vedrete il led rosso sempre acceso mentre quello verde lampeggerà in corrispondenza di ogni blocco di dati rice-

vuto. Concludendo, vogliamo precisare che file letti da un registratore (sia questo un SPC01 o SPC02) con due memorie non sono utilizzabili con un registratore con tre o quattro memorie e viceversa; inoltre, se tenterete di scrivere un file con più di due tracce su un modulo SPC02, in fase di riproduzione non riuscirete ad accedere alle tracce successive alla seconda. ■

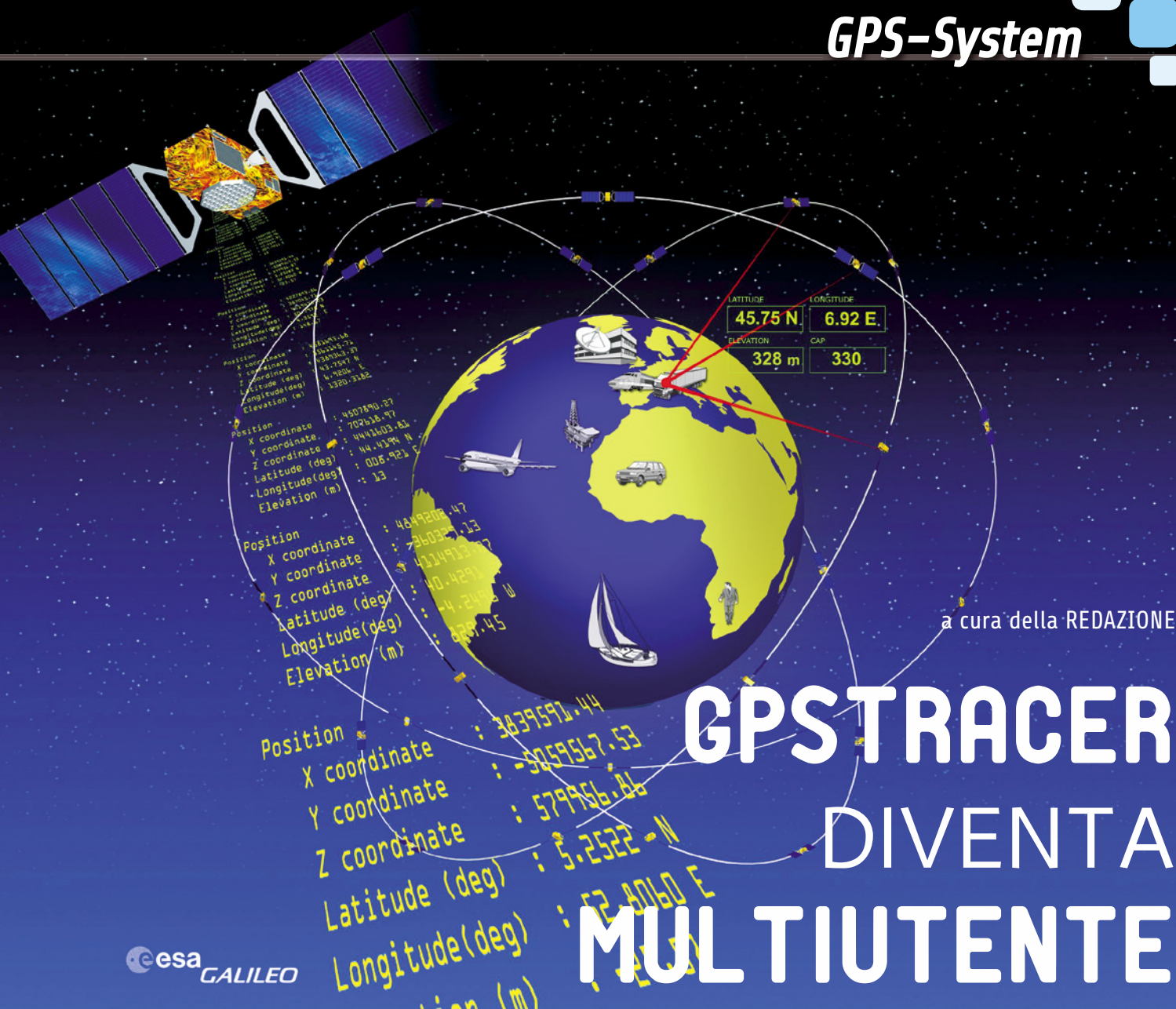


per il MATERIALE

Il modulo vocale SPC02 è disponibile già montato e collaudato al prezzo di 28,00 Euro. Il circuito comprende tutti i componenti e le due memorie da 1 Mbit; non sono compresi, invece, i pulsanti, l'altoparlante ed il microfono. Ricordiamo che la versione base del modulo SPC01 descritto nel fascicolo precedente è disponibile al prezzo di 34,00 Euro; la relativa demoboard (SPC-DK) costa invece 22,00 Euro. Il software di gestione di entrambi i moduli - descritto in questo numero - può essere scaricato gratuitamente dal sito della rivista (www.elettronica.in.it). I prezzi si intendono IVA della compresa.

Il materiale va richiesto a:

Futura Elettronica, Via Adige 11, 21013 Gallarate (VA)
Tel: 0331-799775 - Fax: 0331-792287 - www.futurashop.it



Possibilità di visualizzare più target contemporaneamente e tracciare il percorso sulla base dei dati memorizzati: sono queste le novità della nuova versione di *gpstracer.net*.

Nell'ottobre nel 2005, in occasione della presentazione del localizzatore GPS FT596, abbiamo messo a disposizione dei possessori di tale apparato (oltre ai prodotti G19B e WEBTRAC4S) un sito Web per il tracciamento degli automezzi; il servizio sfruttava le coordinate giunte dal localizzatore con cellulare GSM/GPRS via SMS per ricavare gli indirizzi Web da inserire nel browser ed avere, su pagine del sito di *Tuttocittà*, la posizione di un dispositivo. Nel 2006 *Futura Elettronica* ha sviluppato

GPSTracer, un portale che permette ai proprietari di localizzatori di tracciare il percorso dei veicoli a bordo dei quali sono installati i GPS; per utilizzare il servizio basta registrarsi (bisogna esser clienti *Futura Elettronica*) e comunicare al servizio i dati del cellulare del localizzatore. Quindi, ogni volta che il telefono invia la posizione dell'apparato mediante SMS, il gestore genera una e-mail che viene elaborata da *GPSTracer*; collegandosi al sito *www.gpstracer.net* si poteva seguire sullo schermo del PC il percor-



Fig. 1

so dell'automezzo con a bordo il localizzatore. In un primo tempo il servizio, del quale vi abbiamo parlato nel fascicolo 107, si limitava alla visualizzazione di una posizione alla volta e relativamente a un veicolo. Da poco più di un mese il portale GPSTracer si è evoluto e consente, appoggiandosi alla cartografia Google Maps, di visualizzare contemporaneamente la posizione anche di più di un localizzatore per utente: il tutto gratuitamente (a meno del costo legato alla SIM presente nell'unità remota).

Per accedere al servizio occorre, come al solito, registrarsi e ottenere la password, quindi impostare i dati di uno o più localizzatori dai quali ricevere le informazioni. I localizzatori possono inviare le informazioni sul posizionamento mediante SMS o e-mail, usando l'apposito servizio messo a disposizione dai gestori, di e-mail da telefono cellulare; sullo schermo, nella piantina, l'utente può visualizzare una posizione, un tracciato già percorso, oppure lo spostamento in real-time. In realtà, non si può propriamente parlare di tempo reale, perché i limiti del sistema di comunicazione tra localizzatore e sito Web impongono

intervalli di qualche secondo tra una posizione e quella successiva. Il servizio permette all'utente di definire l'intervallo di aggiornamento della posizione nella modalità real-time; per intervallo di aggiornamento si intende il tempo che passa tra due elaborazioni di dati consecutive. Il tempo corrispondente può essere scelto fra un secondo e alcuni minuti; come detto, nella pratica non conviene scendere sotto l'intervallo che normalmente è richiesto dai sistemi di comunicazione. A titolo d'esempio, tenete

presente che se si inviano i dati via SMS, ora che il messaggio viene composto e arriva a destinazione, possono passare anche qualche decina di secondi; se si adotta l'invio tramite il servizio e-mail, anche in questo caso, per bene che possa andare devono trascorrere almeno 20÷30 secondi prima che la e-mail venga smistata dalla rete cellulare GPRS e inviata al server del sistema. Perciò, dato che lo scopo di GPSTracer è quello di tracciare percorsi da esaminare in un secondo tempo o seguire veicoli che si spostano in maniera relativamente lenta o che comunque non possono percorrere troppa strada tra un invio di dati e l'altro, un intervallo di aggiornamento ragionevole è di una trasmissione al minuto.

LE FUNZIONI DEL SISTEMA

Vediamo di esaminare da vicino le funzionalità offerte dal servizio; per prima cosa va detto che vi si accede puntando, dal browser, all'indirizzo www.gpstracer.net/indexnew. La pagina Web che si apre (Fig. 1) è quella del login e propone due caselle per l'identificazione dell'utente

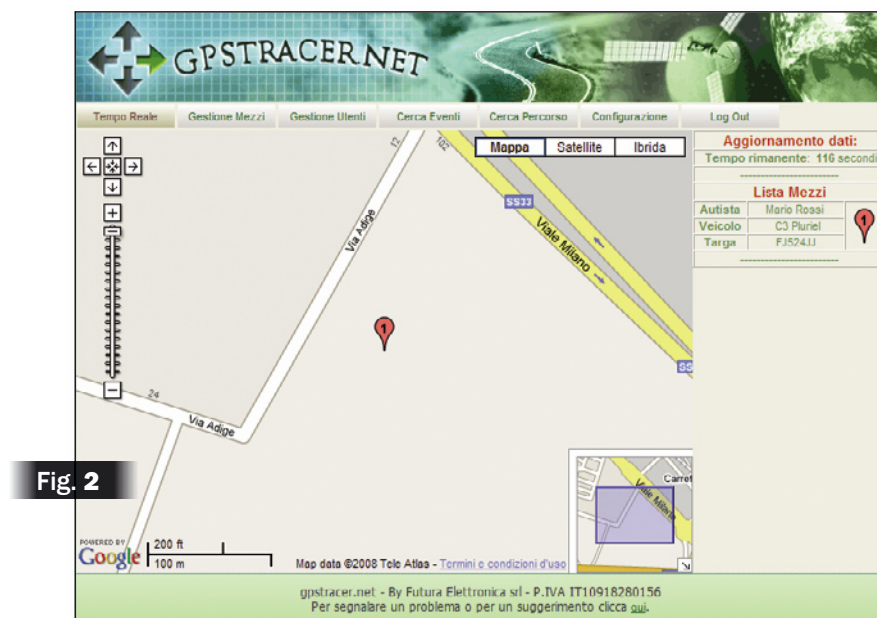


Fig. 2



Fig. 3

(account utente e relativa password); chi non ha ancora un account, deve fare clic nell'apposito link, dal quale ha accesso al form di registrazione. Una volta inseriti utente e password, si può accedere ai servizi.

La pagina iniziale che segue il login è composta da alcune schede, corrispondenti ciascuna ad una delle funzionalità. La scheda *Tempo reale* (Fig. 2) avvia la funzione di visualizzazione del percorso di un localizzatore attualmente in funzione e collegato al sistema (che invia le coordinate); man mano che esso manda le coordinate dei punti in cui si trova allo scadere del tempo impostato, l'interfaccia grafica aggiunge la posizione corrispondente e viene disegnato il percorso. L'aggiornamento dipende, come detto, dal tempo impostato. In alto a destra, una casella mostra, istante per istante, quanto manca al successivo aggiornamento. Sempre a destra, un riquadro riepiloga i veicoli registrati per l'utente cui si riferisce la sessione di lavoro aperta; in esso è possibile scegliere, se sono più d'uno, quello di cui si vuole che venga mostrato il tracciato o la posizione.

La scheda *Gestione mezzi* (Fig. 3) raggruppa le opzioni riguardanti gli automezzi e quindi i localizzatori corrispondenti: in essa si trovano le informazioni che

l'utente ha inserito per definire un veicolo (nome del conducente, targa, modello e tipo di localizzatore: ad esempio WEBTRAC4S) e il codice IMEI del cellulare utilizzato dal localizzatore stesso per inviare i dati relativi alle coordinate fornite dal ricevitore GPS. Sempre dalla scheda *Gestione mezzi*, si può scegliere, per l'automezzo attualmente selezionato, che cosa visualizzare: il comando *Percorso mezzo* traccia il percorso compiuto dall'ultimo reset dei dati, percorso che viene costruito sulla base dei punti di localizzazione forniti in ordine cronologico dal localizzatore a bordo del veicolo. Facendo clic sul comando *Modifica* è possibile modificare l'attuale impostazione del parametro corrispondente; ad esempio, se si fa clic su *Modifica* accanto alla targa del veicolo, si accede a una finestra di dialogo che permette di

cambiare il numero di targa del veicolo relativo al localizzatore cui si riferisce la sessione di lavoro aperta.

Particolarmente interessante è la sezione *Eventi mezzo* (cui si accede facendo clic sull'omonimo comando della scheda *Gestione mezzi*) perché consente di visualizzare, in ordine cronologico, gli eventi corrispondenti al localizzatore del veicolo; per eventi si intende tutti i dati ricevuti dal localizzatore e non necessariamente informazioni sulla posizione. I dati trasmessi possono riguardare una posizione rilevata dal GPS e trasmessa autonomamente dal localizzatore o una inviata su richiesta del servizio *GPStracer*; oppure, per i localizzatori provvisti di ingressi di allarme, la segnalazione che un ingresso è stato attivato.

Nella sezione di destra della finestra di dialogo *Eventi mezzo* (Fig. 4), trovate un modulo di ricerca nel quale è possibile effettuare

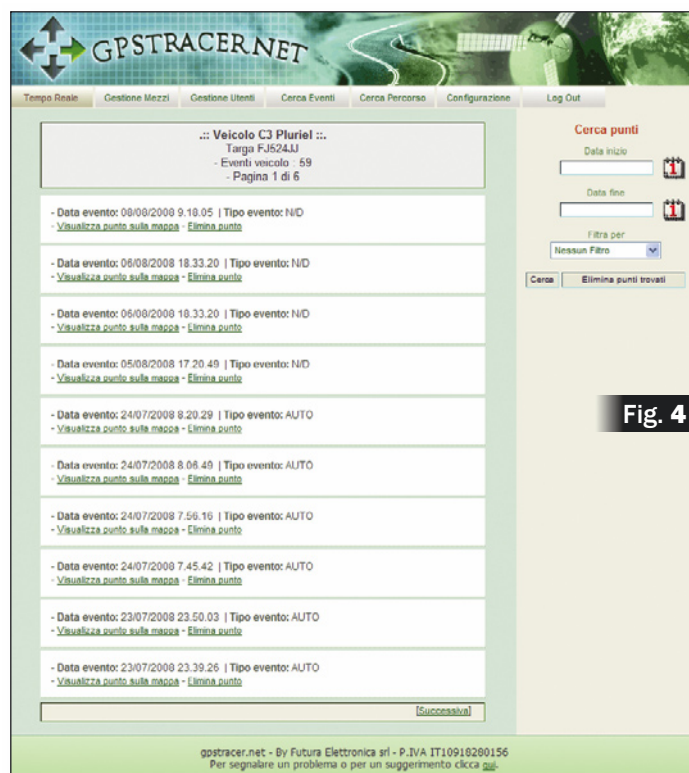


Fig. 4



Fig. 5

la ricerca degli eventi entro un intervallo temporale definibile con le caselle *Data inizio* e *Data fine*; i dati estratti possono essere filtrati applicando eventualmente un filtro di quelli selezionabili dal menu a tendina che si apre facendo clic nella casella *Filtra per*. Ad esempio, è possibile scegliere i soli dati di posizionamento corrispondenti alle variazioni di posizione, ovvero quelli relativi agli spostamenti (vengono esclusi i punti uguali consecutivi) oppure i dati trasmessi quando il contatto di allarme del localizzatore è aperto o chiuso. L'ultimo comando della finestra *Gestione mezzi* è *Esporta coordinate*: facendovi clic, si apre una finestra di dialogo che propone la creazione e l'apertura o il salvataggio dei dati sulla posizione corrispondenti al localizzatore

cui si riferisce la sessione di lavoro, in un file *.xml*.

Oltre alle funzionalità di visualizzazione, tracciatura percorsi e gestione dei localizzatori abbinati al servizio, la pagina Web principale contiene funzioni riguardanti la gestione degli utenti abilitati ad accedere al servizio stesso; la scheda *Gestione utenti* (Fig. 5) permette di aggiungere nuovi utenti o rimuovere quelli esistenti, così da ampliare le possibilità di accesso. Per ogni localizzatore o gruppo di localizzatori, l'utente abilitato può definire altri account che possono accedere al servizio; può, altresì, cambiare la propria password,

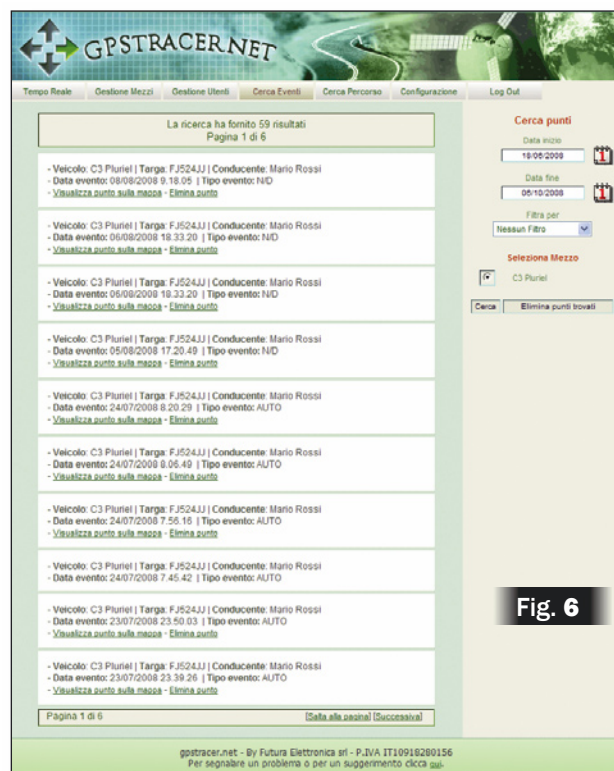


Fig. 6

laddove ritenga che quella in uso non sia più affidabile. La scheda *Cerca evento* (Fig. 6) contiene i comandi della sezione di destra della finestra di dialogo accessibile con il comando *Eventi mezzo* (*Gestione mezzi*) già descritto: sostanzialmente è un doppiante di tale sezione e serve ad estrarre direttamente le informazioni sul posizionamento corrispondenti al localizzatore, eventualmente filtrate. Offre in più la possibilità di selezionare direttamente il veicolo, ovvero il localizzatore; la casella corrispondente manca in *Eventi mezzo* perché in tale sezione è superflua, dato che da questa si accede solo dalla finestra di dialogo relativa a un certo mezzo, quindi vuol dire che i dati sono quelli di uno specifico veicolo. La scheda *Cerca percorso* (Fig. 7) si avvale delle mappe di Google per tracciare un percorso con i punti scelti nelle caselle della sezione a destra della finestra di dialogo corrispondente; la scelta dei punti di posizione si effettua per data e, volendo, può essere



Fig. 7

subordinata ad un filtro (selezione per tipo di evento). I dati estratti, per il veicolo scelto con l'apposito pulsante di opzione in basso a destra, disegneranno sulla mappa il tracciato del caso, indicando eventualmente in quali punti si è verificato un evento quale l'apertura o la chiusura del contatto di allarme.

Particolare interessante è il fatto che, utilizzando il servizio *Google Maps*, è possibile visualizzare il tracciato su una carta geografica, su una mappa stradale o sulla vista da satellite, nelle zone per le quali *Google* la rende disponibile.

In ultimo, vediamo le funzioni connesse alla scheda *Configurazione* (Fig. 8): facendo clic sull'omonima linguetta, si accede alla pagina Web www.gpstracer.net/Reg/Schede_U.asp, nella quale sono riportate le istruzioni di configurazione dei localizzatori GPS con cellulare GSM/GPRS da noi pubblicati; in essa trovate in forma essenziale ma efficace, le informazioni inerenti all'impostazione dei parametri di funzionamen-

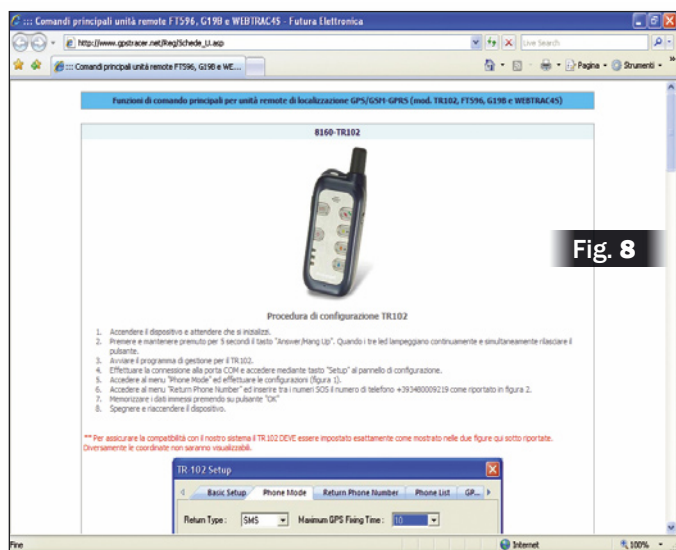


Fig. 8

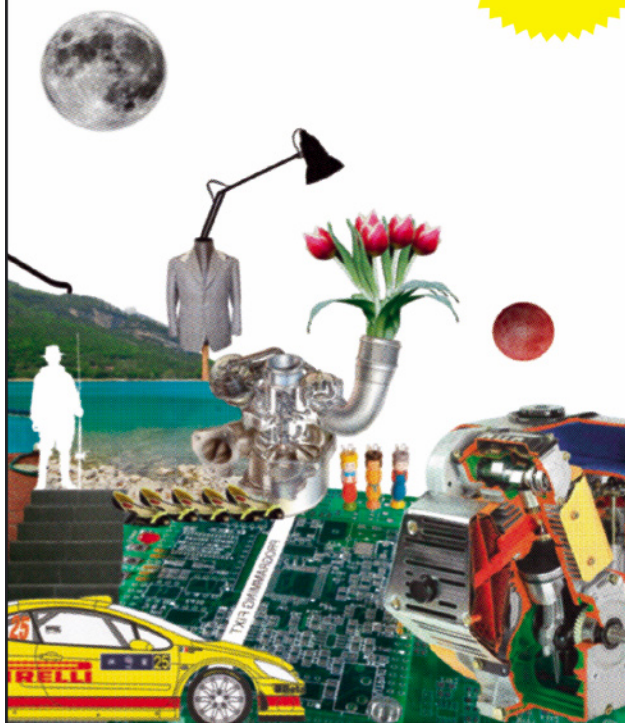
to dei dispositivi mediante le specifiche routine su PC. Tale funzionalità è l'ideale quando si debbano modificare parametri quali quelli riguardanti il funzionamento del ricevitore GPS, la cadenza e la modalità di trasmissione dei dati (polling o invio automatico -Autoreport-) il numero telefonico ecc. In ogni momento si può uscire dall'interfaccia utente del servizio facendo clic sulla linguetta della scheda *Logout*; fatto ciò, per rientrare occorre ripetere il login dalla pagina iniziale www.gpstracer.net.

la seconda luna
premio nazionale alle passioni

15000 EURO PER LA TUA PASSIONE

La seconda luna è il concorso che premia la tua passione, qualunque essa sia.

Il bando completo su:
www.lasecondaluna.it



Comune di Laives
Stadtgemeinde Leifers



Fondazione
Cassa di Risparmio
di Bolzano

STOP AI BLACKOUT!



Sistema di monitoraggio e limitazione automatica del consumo per impianti a 220 Vca, fino a 6 kW. Legge continuamente la potenza assorbita dagli elettrodomestici connessi all'impianto elettrico e, in caso di superamento della potenza massima, secondo una precisa priorità, disattiva quelli di cui si può fare a meno. Prima puntata.

di ROBERTO PRESTIANNI

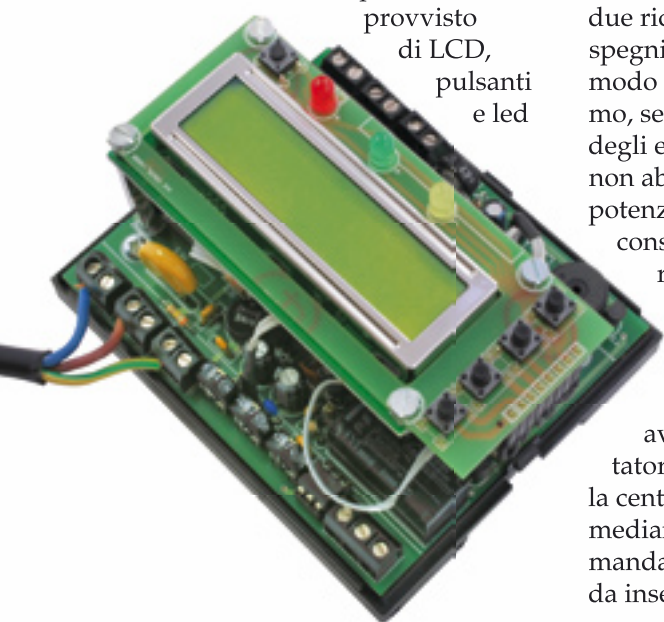
Quante volte veniamo bruscamente disturbati da un blackout? Certamente, specie d'estate, quando il consumo è particolarmente elevato a causa dell'uso massiccio di condizionatori d'aria, non è infrequente vedere andar via la corrente all'improvviso. I blackout dovuti a inconvenienti tecnici del nostro gestore elettrico sono, per fortuna, relativamente brevi e sporadici; più frequenti sono quelli che procuriamo noi stessi quando consumiamo più potenza di quanta ne possiamo assorbire per contratto. Può accadere ad esempio che, mentre stiamo

usando la lavatrice, ad un certo punto accendiamo pure il forno elettrico; se ci aggiungiamo elettrodomestici quali phon, stufette, condizionatori, tutti i dispositivi per l'illuminazione e le apparecchiature elettroniche, ecco che il contatore ha da dire la sua e decide di sconnetterci l'impianto, soprattutto se è di quelli elettronici, che hanno poca tolleranza. Per evitare ciò, vi proponiamo l'antiblackout descritto in queste pagine: si tratta di un controllore elettronico che si preoccupa di organizzare i consumi di elettricità e verificare che non venga superato il limite

che fa scattare l'interruttore del contatore. Quando chiedete troppa corrente, disconnette automaticamente uno o più utilizzatori, per fare in modo che il consumo rientri nei limiti ammessi; per non fare tagli indiscriminati, grazie ad una gestione basata sulle priorità di utilizzo degli elettrodomestici decide cosa deve spegnere per primo.

Il sistema consta di una centralina, installata tra il salvavita ed il resto dell'impianto elettrico, che misura l'assorbimento e, se è il caso, comanda via radio uno o più ricevitori che commutano l'alimentazione dei carichi che si intende staccare dalla rete. Dunque, gli utilizzatori che considerate opzionali e di cui ritenete di poter fare a meno, vanno alimentati attraverso i moduli attuatori, in modo che, all'occorrenza, la centralina possa disattivarli. Il sistema comanda gli attuatori mediante segnali radio a codifica Motorola, cosa che permette di abbinare ad esso qualsiasi ricevitore da radiocomando basato su questo tipo di codifica, purché la sua parte radio operi alla frequenza prevista. Si presenta come un modulo per barra DIN

provvisto di LCD, pulsanti e led



frontali, alloggiato in un box plastico a 6 posti; l'attuatore (Smart-Rx) è un box con spina integrata, una presa volante per il collegamento dell'apparecchio elettrico da controllare, un'antenna per la ricezione dei comandi emessi dalla centralina e poi ancora led e pulsanti frontali. Per spiegare come funziona il sistema, supponiamo che tra gli utilizzatori di casa alcune lampade, un PC, una lavatrice ed un phon sono quelli da sconnettere in caso di eccessivo consumo; in particolare, stabiliamo un ordine con cui spegnerli: 0-phon, 1-lavatrice, 2-lampade, 3-PC.

I livelli di priorità previsti sono 7, ossia da 0 a 6.

Dunque, se a un certo punto i quattro apparecchi sono alimentati e operativi e la centralina registra un consumo di energia eccessivo, la stessa emette il comando di spegnimento prima del livello zero (il phon), e se ciò non dovesse essere sufficiente passerà anche al livello uno (spegnimento lavatrice); verrà eseguita un'altra lettura e se, malgrado il distacco dei carichi 0 e 1, la potenza massima risulterà superata, anche gli attuatori del livello due riceveranno il comando di spegnimento (le lampade). In tal modo il PC sarà spento per ultimo, sempre che la disattivazione degli elettrodomestici precedenti non abbia riportato il consumo di potenza entro il valore massimo consentito. Può però accadere che l'intervento della centralina, e quindi la disattivazione di tutti i livelli, non sia sufficiente, allora un segnale acustico avvertirà che a breve il contatore staccherà la linea. Allora la centralina isola l'impianto mediante un relé pensato per comandare un teleruttore opzionale da inserire nel quadro elettrico.

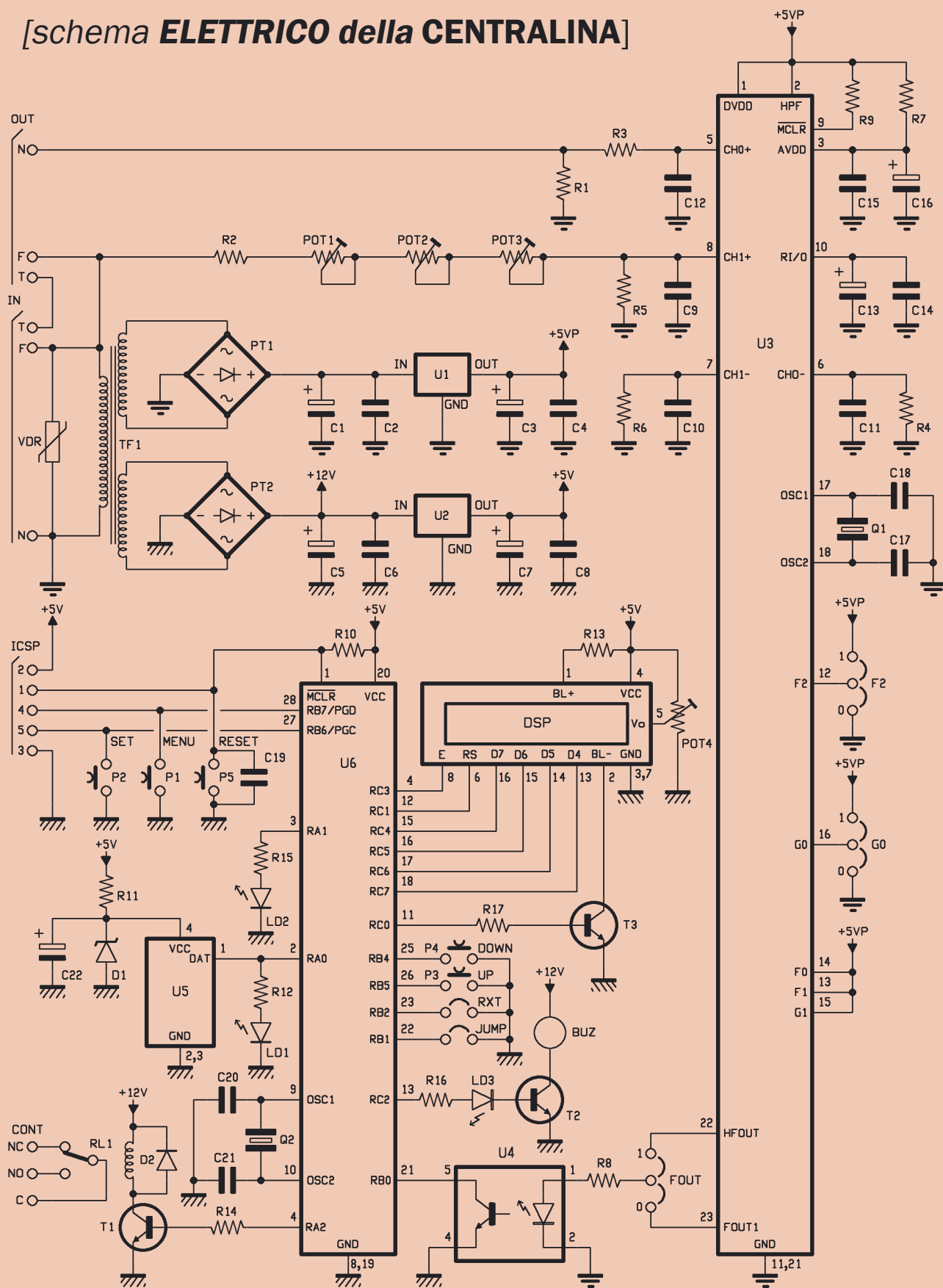
Vediamo ora come il sistema riattiva i livelli e quindi tutti gli elettrodomestici: riportato l'assorbimento sotto il valore limite, la centralina prova a riconnettere gli utilizzatori nell'ordine inverso in cui li ha disconnessi, quindi se ha staccato per ultimo quello di priorità 4 e nel riconnetterlo verifica che l'assorbimento conseguente è nel limite, tenta di ricollegare il dispositivo a priorità 3.

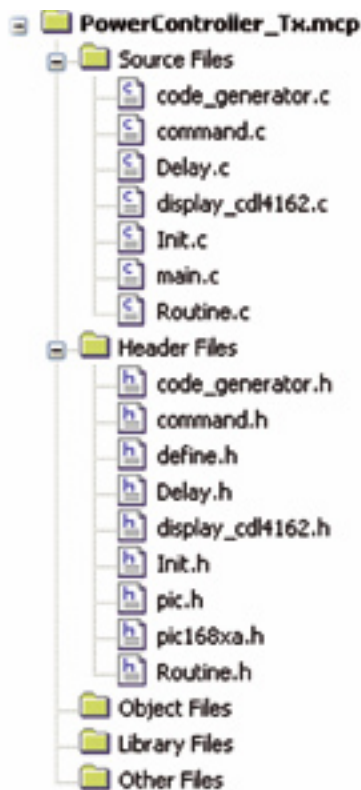
Supponiamo ora che tutti i ricevitori siano in OFF e il relé di spegnimento generale abbia staccato l'impianto, nel tentativo di non far "saltare" il contatore. Dopo circa 30 secondi questo relé viene rilasciato in modo da ridare tensione all'impianto. Viene allora fatta un'ulteriore lettura di potenza e se il sovraccarico perdura, il relé interviene di nuovo, altrimenti la centralina prosegue riattivando l'ultimo livello posto in OFF (a più alta priorità). Adesso, se gli elettrodomestici di questo livello sono stati spenti dall'utente e se il sovraccarico è finito, dopo altri 30 secondi il livello sottostante (nell'ordine di priorità) viene anch'esso riattivato e così fino a riportare il sistema al funzionamento normale: tutti i livelli in ON. In caso contrario, dopo pochissimi secondi quello stesso livello verrà riposto in OFF. La centralina misura la potenza consumata, massima e media; permette, inoltre, di definire il massimo livello di potenza erogabile e la trasmissione per l'abbinamento dei ricevitori.

SCHEMA ELETTRICO

I componenti principali sono il sensore di potenza Microchip MCP3905A e il microcontrollore PIC16F876A. Il primo (U3) si occupa del calcolo della potenza istantanea erogata all'impianto elettrico, leggendo corrente e

[schema **ELETTRICO** della CENTRALINA]





Lista completa dei file che compongono il firmware dell'antiblackout, con sorgenti (.c) e header (.h).

tensione istantanea e calcolandone il prodotto, mentre il secondo (U6) acquisisce serialmente le letture sulla linea seriale, pilota il modulo LCD, il trasmettitore RF, i pulsanti, i led, il relé e tutte le altre parti del circuito che supportano le funzionalità dell'Antiblackout. Il punto di congiunzione tra le due parti lo realizza otticamente il fotoaccoppiatore U4 che, allo stesso tempo, isola galvanicamente il micro dal misuratore di potenza, condizione necessaria a separare la linea elettrica dal resto del circuito che, avendo comandi manuali accessibili dall'utente, deve garantire un elevato livello di protezione. Ciascuna sezione ha un proprio stadio di alimentazione (soluzione anche questa volta alla separazione galvanica) c.c. a 5 V; ognuno è alimentato da un unico trasformatore (TF) a doppio secondario da 9V AC. Oltre alle

uscite a 5 V, una linea a 12 V ricavata dall'uscita del ponte PT2, alimenta il relé e il cicalino BUZ. La linea di potenza di ingresso (FIN, NIN, TIN) è riportata in uscita, interponendo sul neutro una resistenza da 0,001 Ω usata dall'MCP3905 per misurare la corrente alternata diretta all'impianto. Il neutro fa da massa per il circuito del sensore, il che fornisce una sicurezza ulteriore, dal momento che non è la fase ad andare "in giro" per il circuito. La tensione caduta sul resistore da 0,001 ohm viene applicata all'ingresso CH0 dell'MCP3905 (tramite il filtro passa-basso R3/C12) di misura della corrente; il secondo ingresso di misura, CH1, legge la tensione di rete prelevata dalla fase e riferita alla massa (quindi al neutro), ricavata dal partitore formato da R2-POT1-POT2-POT3 ed R5. Il fattore di riduzione, una volta tarato il circuito, è di diverse centinaia di volte e, nel peggiore dei casi (POT1-POT2-POT3 in corto), risulta pari 560 (R2 è da 560 k Ω). Questo comporta una tensione efficace massima (per $V_{in}=230$ Veff) sull'ingresso differenziale CH1, pari a 0,411 Veff e quindi un valore di picco di $\pm 0,580$ V, ben al di sotto del valore massimo ammesso da questo ingresso ($\pm 0,660$ V). Anche la tensione differenziale applicabile su CH0 ha un limite, ed è pari a ± 470 mV/G, dove G è il guadagno del PGA. Nel progetto $G1=1$ e $G0=0$ quindi il guadagno sarà pari ad 8, e così il limite di tensione è di ± 60 mVp. I condensatori C9÷C12 servono a filtrare le tensioni ai capi CH1 \pm e CH0 \pm . L'ingresso CH1 serve per rilevare la fase della tensione, che l'MCP3905 compara con quella della corrente, desunta da CH0. Il clock del sensore è generato impiegando il quarzo X1 e i

due condensatori C17 e C18. La linea F2 (pin 12) va posta a zero tramite il jumper F2, così da assegnare il valore di 27.968,75 Hz alla frequenza di riferimento per l'uscita impulsiva HFc. Questo particolare consente di effettuare letture di potenza istantanea in soli 2 secondi senza perdite di definizione dovute alla conversione numerica. Il jumper Fout, invece, consente di selezionare una delle due possibili uscite impulsive dell'integrato che, in questo progetto, sarà HFOUT. Infine, R7, C15 e C16 realizzano un semplice filtro per il pin Avdd, che alimenta con 5 V la parte analogica del semiconduttore, mentre il varistore VDR protegge da eventuali picchi presenti sulla rete.

Passiamo adesso al PIC: gli impulsi da contare per calcolare la potenza sono applicati al pin RB0/INT, cui giungono indirettamente per mezzo del fotoaccoppiatore U4. I pulsanti per la gestione del menu (MENU, SET, UP, DOWN) sono connessi alle quattro porte RB7/6/5/4, configurate come input e provviste di resistenze di pull-up interne. A due di queste linee fanno capo i terminali del connettore da 6 poli ICSP, necessario per la programmazione In-Circuit. Il reset del PIC all'avvio è ottenuto con R10/C19; quello manuale si opera con l'apposito pulsante. La centralina trasmette i codici Motorola per comandare l'accensione e lo spegnimento dei carichi connessi agli attuatori, modulando in ampiezza una portante radio mediante il modulo ibrido Aurel TX-4M10HA, dotato di antenna integrata e oscillatore da 10 mW, la cui alimentazione (3 Vcc) è ottenuta mediante la rete R11-D1-C22. Ogni volta che avviene una trasmissione, il led giallo LED1 lampeggia per tutto

il periodo. Il distacco completo dell'impianto elettrico avviene ad opera del relé RL1, pilotato dal transistor T1. Il led LD2, pilotato direttamente dalla linea RA1 tramite la resistenza di caduta R15, segnala ciascuna lettura della potenza elettrica con un impulso luminoso. BUZ (un cicalino con oscillatore integrato) e LED3 danno la segnalazione acustica e visiva di sovraccarico. DSP è un display LCD 2 righe e 16 caratteri retroilluminato, pilotato dal PIC con soli 4 bit dati (DB7/6/5/4) oltre le linee E, RS e BL-, quest'ultima portata a massa da T3 ogni volta che DSP deve essere illuminato. La resistenza R13 completa il circuito di alimentazione per la retroilluminazione e POT4 consente la regolazione del contrasto dello stesso display. La sincronizzazione di U6 (il microcontrollore) è realizzata impiegando il quarzo X2 da 16 MHz, e dai condensatori C20-C21.

IL FIRMWARE

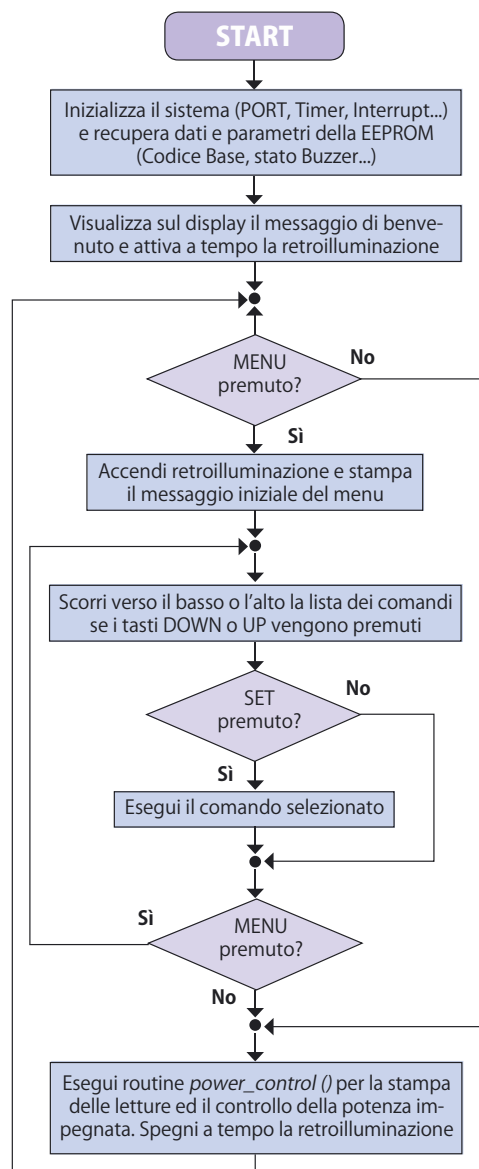
Dal nostro sito è possibile scaricare sia il firmware per il PIC16F876A, sia l'intero progetto software con tutti i file sorgente in linguaggio ANSI C, la cui lista (compresi gli headers e i file ad estensione .h) è illustrata nell'apposita figura.

Il *main.c* contiene sia la *main()* del progetto che la routine di interrupt. La prima realizza sostanzialmente il menu e la chiamata dei relativi comandi: leggendo lo stato dei tasti funzione (MENU, SET, UP, DOWN) fa apparire i 12 comandi e ne consente lo scorrimento verso il basso o l'alto, il puntamento e la selezione. Quando viene selezionato il comando, si richiama ed esegue immediatamente la routine competente. La seconda funzione (routine di interrupt), realizza i conteggi

lunghi (impiegando il TIMER1 con registro da 16 bit) necessari allo spegnimento automatico della retroilluminazione del display e all'attivazione di quest'ultima se uno dei quattro pulsanti viene premuto. Il file *command.c* definisce, invece, tutti i comandi implementati dal sistema, e quindi le relative funzioni: *print_power_mas()* calcola e stampa sul display la potenza massima letta dall'ultimo azzeramento; *print_power_average()* calcola invece la media delle letture di potenza istantanea; *power_control()* realizza il controllo della potenza comandando gli attuatori dei diversi livelli di priorità, così come il relé del sovraccarico, il cicalino e il led rosso che segnala questa condizione (questa funzione verrà meglio esaminata nel seguito). Il comando *active_all_Rx()* consente l'attivazione forzata di tutti gli attuatori (sia gli Smart-Rx che i ricevitori Motorola di altro tipo); *disactive_all_Rx()*, in modo complementare alla precedente, disattiva forzatamente tutti gli attuatori; *print_level_Rx()* assolve, invece, al compito di far visualizzare ripetutamente il livello priorità dai soli Smart-Rx attraverso il loro led verde e, ancora, *reset_system()* consente il reset completo del sistema dopo il ripristino dei valori di default.

Il comando *insert_base_code()* serve per la modifica del codice base e l'inserimento del codice del livello priorità da trasmettere agli attuatori, in fase di codifica con autoapprendimento, mentre *print_base_code()* visualizza su display l'attuale codice base; *buzzer_select()* abilita/disabilita il cicalino di segnalazione del sovraccarico non controllato e *write_power_max()* consente, invece, l'inserimento della massima potenza erogabile senza che intervengano gli attuatori. Il comando

write_priority_max() realizza l'inserimento del massimo livello di priorità gestito dal sistema in caso di sovraccarico; *power_read()* compie la lettura della potenza istantanea (chiamata da più funzioni) e, per finire, *print_power()* stampa la lettura della potenza. Il file *code_generator.c* sovrintende e realizza tutte le trasmissioni dei codici Motorola e contiene le seguenti routine: *init_code_gene-*



Struttura generale del firmware (il diagramma di flusso rispecchia il contenuto del *main-program* del micro).

rator() recupera dalla EEPROM dati del PIC il parametro che definisce la frequenza di emissione dei bit di ciascun codice (*delay_timer0*); *set_freq()* invece, consente la taratura di tale frequenza modificando lo stesso parametro e, inoltre, permette l'acquisizione del codice base e del livello priorità da parte dei ricevitori con autoapprendimento; *trans_code()*

è la routine che implementa la codifica Motorola e supporta tutte le altre funzioni; *code_TX()*, di supporto alla precedente, genera singolarmente i bit three-state e li fa trasmettere; *command_code()* codifica i comandi da impartire ai diversi attuatori (compresi i comandi speciali degli Smart-Rx); *halfperiod()* modifica la frequenza di emissione dei codici e, infine,

write_new_code() consente la definizione di un nuovo Codice Base. Il file **Routine.c** definisce alcune funzioni di supporto di secondaria importanza: *welcome_message()* visualizza sul display il messaggio di benvenuto tutte le volte che si resetta il sistema; *lamp_led()* e *more_lamp()* realizzano rispettivamente un singolo lampeggio veloce ed una serie di

La routine a lato ha come argomento il *flag_menu*, definito tramite una variabile da un byte (unsigned char). Questo, se posto ad '1' dalla funzione chiamante, permette l'interruzione della lettura nel momento in cui viene premuto il tasto MENU. Se posto a '0', la lettura non verrà assolutamente interrotta dalla pressione dello stesso pulsante, e questo è necessario quando l'antiblackout è impegnato nella limitazione della potenza istantanea e, quindi, nella disattivazione e controllo degli attuatori (vedere il primo if(...)). Innanzitutto le variabili di conteggio *power_ist* e *counter* vengono azzerate: la prima è una unsigned short (variabile a doppio byte senza segno) che conteggia, ad ogni ciclo di lettura, il numero di impulsi che si presentano al pin HFout dell'MCP3905A, mentre la seconda serve al conteggio dei due secondi (la durata della lettura). Al più, *power_ist* giungerà ad un valore di poco superiore a 6.000, essendo questo il valore della massima potenza attiva nominale prevista dall'antiblackout. Dopo si ha il reset dei flag INTF e TOIF, rispettivamente i flag usati per la segnalazione di un fronte di discesa sulla linea INT/RBO dell'MCU (ovvero la HFout) e il fine conteggio del TIMER1, adoperato qui per il conteggio di ripetuti intervalli da 16 mS (in tutto *num_sec=125* --> *16mSx125 = 2S*). Dopo queste prime inizializzazioni ha inizio il ciclo while() per il conteggio dei 125 cicli. La prima istruzione che si incontra è l' if(...) prima descritto e a seguire l'if(...)

Listato 1

```
void power_read(unsigned char flag_menu)
{
    /**** Calcolo POTENZA *****/
    power_ist=0;counter=0;           //Reset variabili
    INTF=0;TOIF=0;                   //Reset iniziale Flags
    PS2=1;PS1=1;PS0=1;               //Prescaler TIMER0 1:256
    TMRO=5;num_sec = 125;             //Inizializzazioni per il conteggio di 2 secondi
    /*** Conteggio impulsi da HFout entro 2 secondi ***
    while(counter<num_sec)
    {
        if(flag_menu==1 && menu==0)
        {
            flag_menu=0;             //Reset flag
            power_ist=0;counter=0;    //Reset variabili di conteggio
            PS2=0;PS1=1;PS0=1;        //Ripristino Prescaler TIMER0 a 1:16
            return;                   //Uscita se il tasto MENU è stato premuto
        }
        //Conteggio impulsi di HFout *****/
        if(INTF)
        {
            ++power_ist;              //Incremento conteggio impulsi dal sensore
            INTF=0;
        }
        //Fine conteggio TIMER0 *****/

        if(TOIF)
        {
            ++counter;                //Incremento conteggio per attesa 2 secondi
            TOIF=0;TMRO=5;
        }
    }
    PS2=0;PS1=1;PS0=1;               //Ripristino Prescaler TIMER0 a 1:16
    LED=1;DelayMs(100);LED=0;        //Impulso luminoso LED di fine lettura
    if(power_ist>power_mas) power_mas=power_ist; //Rilevamento potenza massima
    /**** *****/
    //NOTA: Per ogni W la frequenza di HFout è incrementata di 0,5Hz
    //(Esempio: 1.000 W --> 500 Hz)
    //Questo significa che in 2 secondi la lettura registra un numero di impulsi pari
    //alla lettura della potenza (Esempio: 1.000 W --> 1.000 impulsi in 2 secondi)
    /**** *****/
}
```

che interroga lo stato di INTF. Ogni volta che questo flag risulterà settato, un impulso si è manifestato e allora la *power_ist* deve essere incrementata. L'ultimo if(...) invece, testa TOIF che risulterà settato ad ogni fine conteggio; allora la variabile *counter* verrà incrementata. Conclusa la serie di cicli

il led verde pulserà, segnalando il termine della lettura. L'ultima istruzione serve al calcolo della potenza massima registrata, che viene eseguita confrontando la lettura attuale con la potenza massima fino a quel momento registrata (*power_mas*).

Il primo if(...) che si incontra verifica che la lettura (*power_ist*) risulti inferiore a 400 W e, in questo caso, un messaggio precompilato verrà inviato al display. In caso contrario (*power_ist*>400 W) la visualizzazione deve riportare l'esatto valore numerico della potenza istantanea. Se la lettura non è inferiore ai 999 W, la routine farà apparire sul display il numero di kW. Diversamente, sarà visualizzata la cifra intera del numero di W (cifra senza virgola). La stessa funzione consente la visualizzazione di tutti i livelli priorità posti in OFF dal sistema, in occasione di un evento di sovraccarico (vedere l'istruzione *switch(counter)*). Questa routine di visualizzazione viene infatti adoperata dalla *power_control()* quando la potenza massima è stata superata. Ma in altre parti del codice sorgente si trova la stessa porzione di codice relativa alla visualizzazione della potenza istantanea.

questi da parte di un led e, per ultima, la *print_menu_command()* stampa il messaggio di ingresso al menu. I file della *Source List* non ancora esaminati sono l'*Init.c* e *Delay.c*: il primo definisce la *Init_system()*, la routine per la configurazione iniziale di tutti i moduli del microcontrollore (PORT, TIMER, oscillatore, Interrupt, ecc.); il secondo contiene invece le funzioni generanti i ritardi. Nello specifico, *DelayMs()* realizza un ritardo multiplo di 1ms; *DelayMss()*, identica alla prima, viene utilizzata dalla sola routine di interrupt; *Delay100Us()* genera un ritardo multiplo di 100 µs, mentre *Delay1S()* conta multipli del secondo. Il file che contiene tutte le routine di gestione del display CDL4162 (2 righe-16 caratteri) è *display_cdl4162.c* che, in questo contesto, non esaminiamo. Per concludere la descrizione dei ruoli di tutti i file del progetto, vediamo adesso a *Header Files* che raccoglie tutti i file header associati a quelli .c appena esami-

```
void print_power(unsigned char *string)
{
    if(power_ist<=400)          //Potenza inferiore a 400W
    {
        strncpy(string, "Potenza < 400W ",16);LCD_Print_string16(string,1); //RIGA 1
    }
    else                        //Potenza superiore a 400W
    {
        if(power_ist>=1000) //Visualizzazione numero di 'kW'(oltre 999W)
        {
            calc2=power_ist-(power_ist/1000)*1000;calc1=power_ist/1000;
            if(calc2<100)
            {
                if(calc2<10)
                {
                    if(calc2==0) sprintf(string,"Power = %1d,000 KW", calc1);
                    else sprintf(string,"Power = %1d,00%1d KW",calc1,calc2);
                }
                else sprintf(string,"Power = %1d,0%2d KW", calc1,calc2);
            }
            else sprintf(string,"Power = %1d,%3d KW", calc1,calc2);
            LCD_Print_string16(string,1); //Scrittura RIGA 1
        }
        else //Visualizzazione numero di 'W' (fino a 999W)
        {
            sprintf(string,"Power = %5d W ", power_ist);LCD_Print_string16(string,1); //RIGA 1
        }
    }
    strncpy(string, "LIV.OFF: ",16);LCD_Print_string16(string,2); //RIGA 2-->Stampa livelli OFF
    for(counter=0;counter<level;counter++)
    {
        LCD_Locate(2,9+counter);
        switch(counter)
        {
            case 0:LCD_Putc('0');break;
            case 1:LCD_Putc('1');break;
            case 2:LCD_Putc('2');break;
            case 3:LCD_Putc('3');break;
            case 4:LCD_Putc('4');break;
            case 5:LCD_Putc('5');break;
            case 6:LCD_Putc('6');break;
            default:break;
        }
    }
}
```

nati, ad eccezione di: *pic168xa.h*, *pic.h* e *define.h*. Il primo definisce nomi e indirizzi di registri, porte e moduli vari del PIC16F876A (U6), mentre il secondo è un file di supporto. Il terzo invece definisce tutte le costanti del progetto: indirizzi EEPROM, pseudonimi delle varie porte, valori di parametri di funzionamento e stringhe di testo del menu. Per semplicità espositiva non sono stati specificati gli argomenti e il tipo di variabile restituita da ciascuna routine. Date ora uno sguardo al flow-chart, che, grosso modo, rispecchia il contenuto della *main()*. All'accensione del microcontrollore il programma al suo interno esegue la *Init_system()* che provvede alle configurazioni dei

registri e dei moduli interni e, in particolare, assegna un valore logico alle porte di I/O collegate ai dispositivi esterni quali led, cicalino, relé, modulo RF. Una volta caricati alcuni dati e parametri prelevati dalla EEPROM, viene visualizzato il messaggio di benvenuto. A questo punto se si preme il tasto l'utente accede al menu dei comandi, che si presenta con un messaggio di benvenuto; dopo circa 20 secondi viene eseguita la *power_control()*, che effettua la lettura della potenza istantanea e ne mostra il valore sul display. Una volta dentro il menu, l'utente, premendo il pulsante DOWN, vedrà apparire i primi due comandi, dopodiché potrà scorrere in basso o in alto i comandi premendo UP o


```

void command_code(unsigned char command,unsigned char level, unsigned char *string)
{
    //Caricamento CODICE BASE
    for(counter=0;counter<6;counter++)
    {
        code_temp1[counter]=EEPROM_READ(EEADD_base_code+counter);
        code_temp2[counter]=code_temp1[counter];
    }
    //COMANDI di ATTIVAZIONE/DISATTIVAZIONE Attuatori/Rx
    if(command == ON_level || command == OFF_level)
    {
        switch (level)
        {
            case 0: code_temp1[6]=ZERO; code_temp1[7]=ZERO; break;
            case 1: code_temp1[6]=ZERO; code_temp1[7]=ONE; break;
            case 2: code_temp1[6]=ZERO; code_temp1[7]=OPEN; break;
            case 3: code_temp1[6]=ONE; code_temp1[7]=ZERO; break;
            case 4: code_temp1[6]=ONE; code_temp1[7]=ONE; break;
            case 5: code_temp1[6]=ONE; code_temp1[7]=OPEN; break;
            case 6: code_temp1[6]=OPEN; code_temp1[7]=ZERO; break;
            default: code_temp1[6]=ZERO; code_temp1[7]=ZERO; break;
        }
        //Copia codice per la seconda trasmissione
        code_temp2[6]=code_temp1[6]; code_temp2[7]=code_temp1[7];
        if(command == ON_level)
        {
            code_temp1[8]=ONE; code_temp2[8]=OPEN;
        }
        else if(command == OFF_level)
        {
            code_temp1[8]=ONE; code_temp2[8]=ZERO;
        }
        //Trasmissione codice di Attivazione/Disattivazione per gli Smart_Rx
        trans_code(code_temp2,7);
        DelayMs(250);DelayMs(250); //Ritardo
        //Trasmissione codice di Attivazione/Disattivazione per tutti gl'altri Attuatori
        if(rx_en==0)trans_code(code_temp1,7); //Solo dietro abilitazione del jumper
        //NOTA:questa sequenza è importante perchè l'Rx del progetto legge il comando del
        //primo codice ricevuto
    }
    //COMANDI SPECIALI per Attuatori/Rx
    else
    {
        //***** Visualizza su LED livello priorità *****
        * if(command == print_level)
        {
            code_temp1[6]=OPEN; code_temp1[7]=ONE; code_temp1[8]=ZERO;
            trans_code(code_temp1,7); //Trasmissione codice comando
        }
    }
}

```

La *command_code()* implementa la trasmissione dei comandi verso gli attuatori. La routine ha per argomenti *command*, *level* e **string* (rispettivamente, comando da trasmettere, livello al quale indirizzarlo, e indirizzo di un array per la memorizzazione delle stringhe di testo da visualizzare su display). I comandi implementati sono: *ON_level*, *OFF_level*, *print_priority*, *ON_all*, *OFF_all* e *chg_base_code*. I primi due consentono l'attivazione e la disattivazione di un particolare livello, *print_priority* realizza il comando 9 per la visualizzazione del livello priorità sugli Smart-Rx, mentre *ON_all* e *OFF_all* realizzano l'attivazione e la disattivazione di massa di tutti i livelli (da 0 fino a quello più alto specificato in fase di configurazione iniziale) e, per concludere, *chg_base_code*, che modifica in automatico il codice base degli Smart-Rx tutte le volte che, col comando 7, si modifica il codice base. Le prime linee di codice della funzione consentono il caricamento del codice base nei primi 6 bit delle variabili *code_temp1* e *code_temp_2*, conservati nella EEPROM-dati a partire dalla locazione di indirizzo *EEADD_base_code*. Il doppio codice è dovuto al fatto che, per alcuni comandi, è necessario usare un codice per gli Smart-Rx e un altro per i comuni ricevitori. Ed anche perché, per mantenere la compatibilità con gli RX bicanale, bisogna emettere due codici che differiscono per l'ultimo bit. Il codice del comando *ON_level* riporta che, dopo la codifica dei bit 7 e

DOWN. Riga dopo riga viene puntato uno dei comandi e, una volta giunti a quello desiderato, agendo su SET lo stesso comando verrà selezionato ed eseguito. Al termine della routine, il flusso del programma ritorna alla *main()*, che chiamerà subito dopo la *power_control()*. Da osservare che, ogni volta che si rilascia il tasto MENU, automaticamente si esce dalla selezione dei comandi e si inizia un ciclo di letture di potenza. In caso di sovraccarico, una sezione della *power_control()* sarà attivata ed eseguita onde comandare gli attuatori. Prima della descrizione di questa importante routine si esaminino *Listato 1* e *Listato 2*, che riportano rispettivamente la funzione per

la lettura della potenza istantanea *power_read()*, e quella per la visualizzazione di tali letture, la *print_power()*. Quanto alla routine *power_control()*, se sull'impianto elettrico non si verifica sovraccarico, realizza solamente la lettura e la visualizzazione sul display della potenza attiva istantanea. Nel caso in cui, invece, si supera la massima potenza, la routine adotta tutte le misure necessarie affinché il consumo dell'impianto rientri nel limite consentito. Dunque, per tutto il tempo in cui l'insieme degli apparecchi elettrici accesi richiede una potenza totale superiore al valore massimo, questa routine rimarrà aperta e continuamente operativa. Inizialmente tutti i livelli prio-

rità saranno in ON (tutti i carichi alimentati), e dopo una prima lettura della potenza istantanea e l'inizializzazione di alcune variabili, la lettura (*power_ist*) viene confrontata col valore della massima potenza precedentemente inserito (*power_max*). Se quest'ultimo non è stato superato, la routine ha termine e il flusso del programma ritorna alla *main*, altrimenti si controlla se l'ultimo livello posto in OFF (inizialmente nessuno lo sarà) è inferiore a quello massimo, ovvero il livello di priorità più alto. In caso affermativo, il livello a più bassa priorità ancora in ON viene disattivato, con l'intento di limitare da subito il consumo di energia. Se invece tutti i livelli risultano

8 (`code_temp1[6]` e `code_temp1[7]`) in base al valore di level (il livello interpellato), il bit 9 in `code_temp1` sarà '1' e in `code_temp2` sarà 'H'. Il primo attiva il canale 1 del ricevitore 2CH, mentre l'altro il canale 2 (anche dello Smart-Rx). Stesso discorso per il comando OFF_level, dove il bit 9 assumerà i valori '1' e '0'. La trasmissione dei due codici avviene con uno scarto di tempo di 0,5 secondi, onde consentire, ad eventuali ricevitori bicanale, la commutazione ora del primo, ora del secondo canale. Le ultime linee del codice riguardano la visualizzazione del livello priorità assegnato a tutti gli Smart-Rx; viene impiegato un codice speciale avente, oltre i primi 6 bit del Codice Base, gli ultimi 3 bit pari a "H-1-0". Se in vicinanza di uno Smart-Rx (avente lo stesso codice Base), si preme il pulsante di trasmissione di un radiocomando configurato con quel particolare codice speciale, si vedrà il led verde dell'attuatore lampeggiare un numero di volte pari al suo livello di priorità. Il jumper può essere Aperto o Chiuso (assente o presente). I valori in grassetto sono quelli preferenziali. G0 va a '0' se il dispositivo deve essere installato in impianti da 4,5 kW o meno, altrimenti va ad '1'. F2 va a '0' affinché U3 lavori alla massima frequenza. Fout deve essere in basso per selezionare HFOUT. RXT deve essere chiuso affinché la centralina possa comandare tutti i ricevitori Motorola (e non solo gli Smart-Rx).

in OFF, l'antiblackout opera il distacco forzato dell'impianto elettrico. Qualora la variabile di conteggio `flag_power_max` risulti uguale al valore definito con `PMX_mess_num` (numero di volte che il messaggio di avvertimento deve essere visualizzato prima dell'intervento forzante), il relé viene tenuto azionato per 30 secondi. Dopo questa attesa il sistema restituisce l'alimentazione all'impianto (relé OFF) per verificare il perdurare del sovraccarico. Se questo dovesse ancora essere presente, il ciclo si ripete con i messaggi di avvertimento e la nuova attivazione del relé. Diversamente (`power_ist <= power_mas`) il sistema tenta di riattivare uno alla volta gli attuatori di tutti i li-

Tabella 1 - I jumper della centralina; in grassetto, le impostazioni predefinite. G0, va su 0 per impianti fino a 4,5 kW e su 1 per potenze da 4,5 a 6 kW. RXT decide la compatibilità con i ricevitori.

JUMPER	1	0	Aperto	Chiuso
G0	G=16	G=8	-	-
F2	HFc=219,51Hz	HFc=27,968,75Hz	-	-
Fout	Fout=HFOUT	Fout=Fout1	-	-
RXT	-	-	Soli Smart-RX	Tutti gli RX

velli, a partire da quello a più alta priorità (vedere la parte sinistra del flow-chart). L'attuatore riattivato viene mantenuto in ON se `power_ist <= power_max`, altrimenti viene nuovamente posto in OFF.

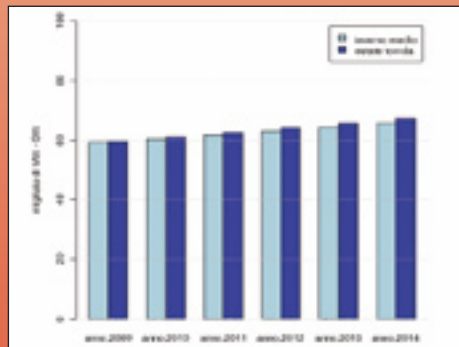
REALIZZAZIONE PRATICA

Gli stampati necessari alla preparazione della centralina antiblackout sono due: quello principale, dove va montata la maggior parte dei componenti e quello a mascherina, che andrà sovrapposto al modulo LCD e sul quale andranno montati tutti i micropulsanti e i led.

Per il c.s. base scaricate dal nostro sito Internet il master che userete per ottenere la pellicola per la fotoincisione. Lo stampato è a doppia faccia, quindi in corrispondenza delle piazzole di interconnessione comuni a entrambi i lati bisogna praticare fori da 0,6 mm in cui inserire corti spezzoni di filo da stagnare da entrambi i lati.

Dal momento che la centralina viene attraversata dalla corrente di tutto l'impianto elettrico (fino a circa 27A), bisognerà stagnare per bene le piste di fase e neutro, in modo particolare quel segmento del Neutro che collega la morsettiera allo shunt. I primi componenti ad essere montati saranno l'MCP3905A (U3) e la resistenza R1. Per il chip bisognerà prestare la massima attenzione, dal momento che i 24 pin sono molto vicini tra loro. Quasi tutte le resistenze vanno montate in verticale ad eccezione di R2-R3-R11, da inserire per prime sul c.s. insieme ai diodi. Il corpo metallico del regolatore U2 andrà

abbassato fino al contatto con la basetta. Tutti i led andranno montati sul secondo c.s., così come i micropulsanti; a questo proposito, poiché i terminali dei tasti MENU, SET, UP e DOWN sono adiacenti, converrà utilizzare un unico strip maschio da quattro pin ed uno da due per i terminali di massa (da montare parallelamente). Così facendo, risulterà più agevole la connessione dei pulsanti al cavetto piatto dell'altra basetta. Per quanto riguarda la connessione del modulo LCD, su questo (dal lato opposto al display) e sul c.s. andrà saldato un connettore strip femmina da 16 pin (alto circa 8mm). A montaggio ultimato, per verificare che tra le linee di Fase e Neutro non vi siano cortocircuiti, misurate col multimetro la resistenza tra queste due piste. Lo strumento dovrà evidenziare la presenza del solo primario del trasformatore, il quale ha una resistenza di circa 1,4 kΩ. Prima di inserire gli integrati U4 e U6, controllate che la tensione dei regolatori sia di 5V: collegate quindi il circuito alla rete e con molta attenzione misurate con un multimetro la tensione continua presente tra i pin 20 (+) e 8 (-) di U6 e quelli di uscita e massa (destra e centrale) del regolatore U1. Durante questa fase ricordatevi che una parte del circuito è connessa alla rete a 230 volt! Se tutto è ok, montate il microcontrollore e provvedete alla sua programmazione col firmware scaricato dal nostro sito, tenendo conto che lo strip femmina ICSP presente sul c.s. riproduce fedelmente la piedinatura del programmatore



Blackout: oggi peggio di ieri

Qual è la causa dell'aumento dei blackout? Il problema sta nella rete di trasmissione e di supervisione e controllo in AT ed MT, alla quale l'apertura del mercato dell'elettricità ha chiesto di essere il veicolo tramite cui i produttori, ovunque dislocati, possono dare corrente a chiunque e ovunque. Il fatto è che per limitare perdite e costi degli impianti, la corrente immessa in rete deve essere consumata relativamente vicino. Se mancano 100 MW al sud-Italia e ne comperiamo dalla Francia anche 200, il problema resta, perché il trasporto per centinaia di km comporta grosse perdite e spese per le linee. Ecco quindi che se anche sulla carta il sistema elettrico nazionale ha abbastanza energia, può capitare che localmente quella che si può sfruttare sia insufficiente perché mal distribuita. Ed ecco allora il verificarsi dei blackout. Alla luce degli aumenti stimati dei consumi negli anni a venire (il grafico di Terna mostra le stime fino al 2014). La soluzione che si cerca è duplice: un sistema automatico di coordinamento in grado di monitorare, a livello locale, la disponibilità di elettricità di tutti i produttori ma anche di operarne l'immissione in rete in base al fabbisogno, ovvero un piano di distribuzione sul territorio dei punti di immissione in funzione dei consumi locali.

Microchip PicKit-2 (il pin 1-Vpp è rivolto verso il trasformatore). Il resistore R1 è uno shunt da 0,001 ohm per multimetri; se non lo trovate in commercio, potete realizzarlo piegando ad "U" un pezzo di filo in rame nudo della sezione di 0,6 mmq, lungo 4 cm. Qualora intendiate impiegare l'antiblackout in impianti da 6 kW, bisognerà cortocircuitarlo con un ricco ponticello di lega saldatore, così da abbassare a circa 0,75 mΩ la resistenza totale. In tal caso è necessario porre a 5V il pin G0 di U3, così da raddoppiare il guadagno del PGA interno (G=16). Inoltre, sarà bene verificare il valore resistivo ottenuto; allo scopo, collegate una presa volante in uscita (ACout) e ad essa un carico elettrico che determini un assorbimento di 10A. In serie ad uno dei cavi di alimentazione dell'ingresso AC inserite un multimetro disposto alla misura di correnti alternate

con fondo scala di almeno 10 A e, prestando molta attenzione, con un secondo tester leggete la tensione tra il nodo R3/R1 e la massa di U3, dopo aver collegato il dispositivo alla rete. Leggete tensione e corrente contemporaneamente e verificate che il loro rapporto non sia maggiore di 0,8 mΩ. In caso contrario, scollegate tutto e stagnate ancor di più la pista interessata. Per concludere, configuriamo i jumper sulla basetta principale come indicato in Tabella 1 (il ponticello JUMP non si usa). Passiamo ora al c.s. del display, nel quale dovete praticare una finestra rettangolare entro la quale far passare l'LCD. Anche in questo caso trovate la traccia lato rame nel sito www.elettronica.in. Iniziate il montaggio con il connettore strip maschio da 8 poli, che va dal lato saldature: prendete lo strip e spostate la barretta in plastica (quella che tiene assieme

i terminali) quel tanto che basta per poterlo saldare agevolmente. Poi saldate i tre led da 5 mm e i 5 pulsanti in modo che risultino a contatto col circuito stampato. Per evitare che venga premuto accidentalmente, il pulsante di reset deve avere il perno cortissimo. L'ultima saldatura da fare è quella di LED3 (led rosso) che, non avendo in comune la massa con gli altri componenti, deve essere collegato al c.s. base con un cavetto lungo 15 cm.

Prima di inscatolare il circuito, bisogna collegare le due basette usando del flat-cable: ad un'estremità salderete un connettore strip femmina da 8 poli (che andrà connesso al c.s. del display), evidenziando uno dei terminali estremi che sarà quello di massa. L'ordine esatto di questi terminali è: 1-massa, 2-MENU, 3-SET, 4-UP, 5-DOWN, 6-LED1 (giallo), 7-LED2 (verde), 8-(RESET). Tenendo a mente questa sequenza dovete saldare sull'altra estremità del cavetto (e in sequenza) uno strip femmina da 1 polo per la massa, uno da 4 poli per i 4 pulsanti, e poi altri 3 da un polo per LED1, LED2 ed il pulsante di RESET. Per poter eseguire le connessioni dei quattro connettori strip al c.s. base, bisognerà separare i relativi gruppi di fili. Infine inserite lo strip maschio da 16 pin con terminali lunghi sul connettore femmina del modulo LCD e collegate il cavetto piatto da 8 pin precedentemente realizzato, con il terminale di massa (evidenziato col pennarello) verso l'alto. Adesso potete eseguire tutte le connessioni: prima quelle di LED3, poi dei quattro pulsanti, la massa, e per concludere i terminali singoli di LED1, LED2 e RESET (terminale sinistro). A questo punto bisogna preparare il cover-top del box da centralino: su carta

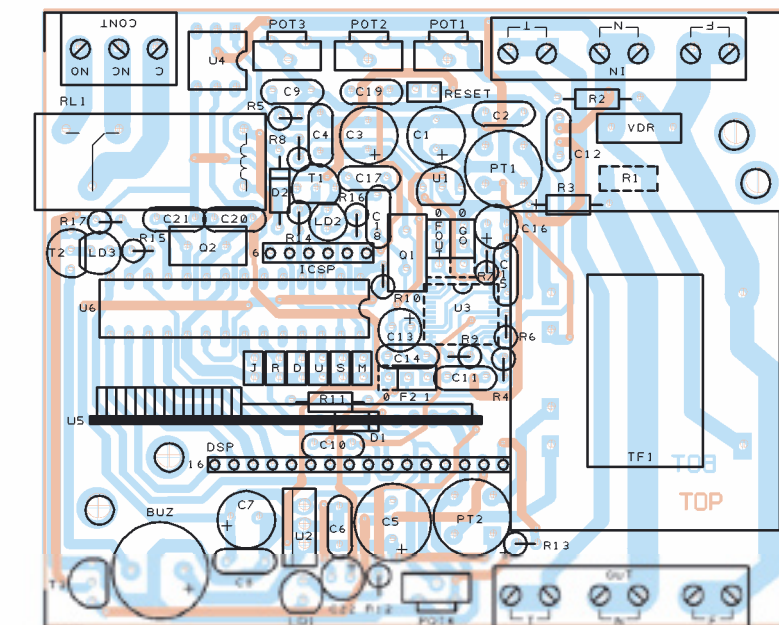
[piano di **MONTAGGIO** della CENTRALINA]

Elenco Componenti:

R1: 0,001 ohm 1W
 R2: 560 kohm
 R3,R4,R5,R6,R9: 1 kohm
 R7: 10 ohm
 R8: 820 ohm
 R10,R14,R17: 10 kohm
 R11: 82 ohm
 R12: 68 ohm
 R15÷R16: 220 ohm
 R13: 22 ohm
 POT1: Trimmer 1 Mohm MV
 POT2: Trimmer 470 Kohm MV
 POT3: Trimmer 47 Kohm MV
 POT4: Trimmer 4,7 Kohm MV
 C1, C3, C7: 100 μ F 25 VL elettrolitico
 C2, C6: 33 nF multistrato
 C9÷C12: 33 nF multistrato
 C4, C8, C14, C15: 100 nF multistrato
 C5: 470 μ F 25 VL elettrolitico
 C13: 10 μ F 25 VL elettrolitico
 C16: 47 μ F 25 VL elettrolitico
 C17,C18,C20,C21: 22 pF ceramico
 C19: 10 nF multistrato
 C22: 47 μ F 6,3 VL elettrolitico
 D1: zener 3 V - 0,6 W
 D2: 1N4007
 PT1, PT2: ponte diodi W06M
 T1, T3: BC547
 U1: 78L05
 U2: 7805
 U3: MCP3905A
 U4: 4N35
 U5: TX-4M10HA
 U6: PIC16F876A-I/SP
 X1: quarzo 3,579545 MHz
 X2: quarzo 16 MHz
 DSP: Display 16x2 (CDL4162)
 TF1: trasformatore 230V/2x9Vac 3VA
 RL1: relè 12V monoscambio 10A
 VDR: varistore 275 Vca
 BUZ: cicalino 12 V c.s. con oscillatore

Varie

- Zoccolo 14+14
- Zoccolo 3+3
- Morsettiera 2 poli (6 pz.)
- Morsettiera 3 poli
- Strip maschio 2 poli (7 pz.)
- Strip maschio 3 poli (3 pz.)



- Strip maschio 4 poli
- Strip maschio lungo 16 poli
- Strip femmina 6 poli

- Strip femmina lungo 16 poli (2 pz.)
- Box 6 moduli DIN-Rail
- Circuito stampato

adesiva, stampate la mascherina di riferimento che riporta le zone da rimuovere; ritagliatela in modo che possa essere fissata (in posizione centrale) sul fondo

della vaschetta che sporge dal cover-top, così da ricavare una guida per la realizzazione di tutti i fori. Aprite la finestra rettangolare, praticate tre fori da 6 mm

per i led e cinque per i pulsanti, usando questa volta una punta da 4,5 mm. Dopo aver pulito il cover-top dai residui di lavorazione, incollate un'altra masche-



Vista d'insieme della centralina: il modulo con il display va collegato alla scheda base mediante flat-cable ed il tutto va inserito in un contenitore per barra DIN opportunamente forato.

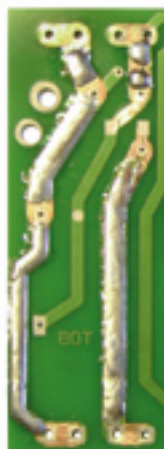
rina adesiva meglio che potete, ritagliandola come il pannello. Concludete il montaggio della centralina avvitando lo stampato base al cover-bottom; chiudete il tutto unendo le due parti del box e facendo affacciare all'esterno i pulsanti, i led e l'LCD.

CALIBRAZIONE E COLLAUDO GENERALE

A questo punto sarà necessario calibrare il sistema in modo che sul display appaia la lettura effettiva della potenza istantanea. Per fare ciò si potrebbe misurare la frequenza del segnale HFOUT con un frequenzimetro e tarare i trimmer in modo che ad ogni watt corrispondano 0,5 Hz; tuttavia, suggeriamo di eseguire delle letture dirette di potenza regolando i trimmer fino a quando sul display non appaia lo stesso valore. L'ideale sarebbe utilizzare un wattmetro per sistemi AC monofase, ma è anche possibile fare ricorso a due strumenti separati (un voltmetro ed un amperometro) moltiplicando i valori letti per ottenere la potenza. Collegate quindi un amperometro per correnti alternate di portata non inferiore a 20 A in serie all'uscita (OUT), e un voltmetro per tensioni AC all'ingresso (IN). Da destra a sinistra (vista dall'alto) i trimmer permettono una regolazione grossolana, media e fine. Alimentate la centralina e attendete il messaggio di benvenuto; premete MENU per dare inizio alla prima lettura. Adesso

collegate un carico da 2.000 W circa e aspettate che il led verde arrivi al secondo lampeggio, così da visualizzare sul display una lettura corretta. Calcolate la potenza disponendo delle letture dei due strumenti e iniziate a ruotare il trimmer di destra (POT1) per una prima regolazione grossolana (ruotando da destra a sinistra si aumenterà il valore sul display), e se è il caso anche quello centrale. Una volta che i due valori si equivarranno nelle unità e nei decimi di kW (le cifre immediatamente a sinistra e a destra della virgola), collegate un ulteriore carico elettrico da 1.800/2.000 W. In questo modo si potrebbe superare la potenza massima consentita dal contatore o, più probabilmente, dall'interruttore magnetotermico della sezione dell'impianto al quale avete collegato la centralina. Siate perciò veloci nella calibrazione con questa ulteriore lettura. In questa occasione sarà il trimmer

Le piste dello stampato che portano dalla morsettiera di ingresso a quella di uscita verso la rete di casa devono essere stagnate per aumentarne la sezione e consentire il passaggio della corrente prevista.



di sinistra (POT3) a dover essere regolato, così da consentire la regolazione fine delle letture. Fatto questo, la centralina è pronta per essere installata.

INSTALLAZIONE E AVVIO

Ora togliete tensione a tutto l'impianto, poi agganciate la centralina alla barra DIN del quadro elettrico di casa. L'entrata (IN) va a valle del salvavita; quanto ai morsetti d'uscita (OUT) vanno al resto dell'impianto elettrico. Se desiderate che la centralina isoli l'impianto elettrico quando il sovraccarico permane, collegate la morsettiera del relé a un contatore da inserire tra l'uscita della centralina e l'impianto.

Dopo aver controllato attentamente il cablaggio, ridate tensione rialzando l'interruttore del contatore. Sul display, apparirà il messaggio di benvenuto il quale terminerà entro 20 secondi, a meno che non si prema il tasto MENU. In seguito ogni due secondi apparirà la lettura aggiornata della potenza attiva consumata dagli elettrodomestici collegati all'impianto. Si noti che quando il sistema è impegnato nella gestione di un sovraccarico, il menu non è disponibile.

MENU COMANDI

Per accedere al menu comandi premere e mantenere premuto il pulsante MENU e, a seguire, il tasto DOWN. Una volta apparse le scritte dei primi due comandi, sarà possibile muoversi sia in alto (con UP) che in basso (con DOWN), puntando di volta in volta il comando adiacente tramite la freccia di sinistra. Per selezionare il comando puntato, premete il tasto SET, dopodiché rilasciate MENU. Il rilascio di quest'ultimo, durante lo scorrimento dei comandi, consente di uscire dal menu e iniziare il ciclo

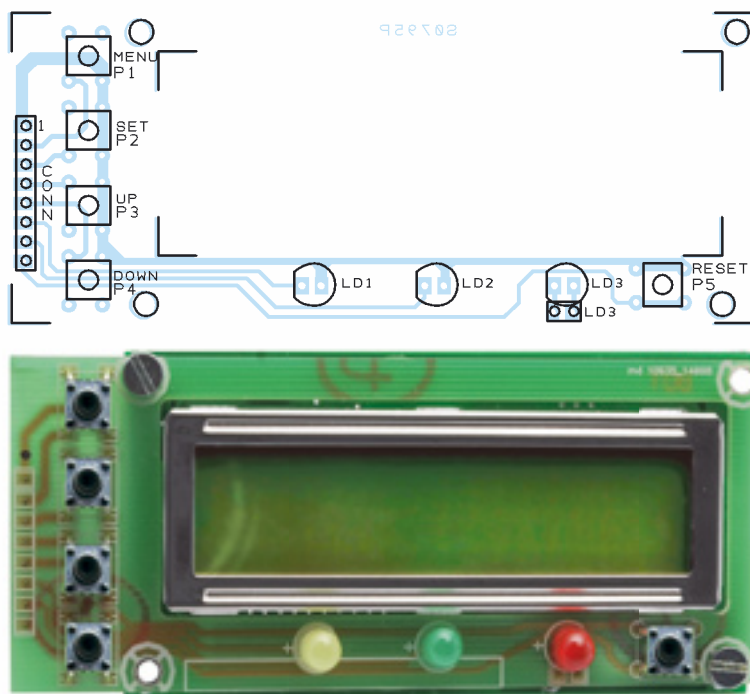
[piano di **MONTAGGIO** della **SCHEDA DISPLAY**]

Elenco Componenti:

- LD1: Led 5 mm giallo
- LD2: Led 5 mm verde
- LD3: Led 5 mm rosso
- P1: Microswitch altezza 3,5 mm
- P2: Microswitch altezza 3,5 mm
- P3: Microswitch altezza 3,5 mm
- P4: Microswitch altezza 3,5 mm
- P5: Microswitch altezza 1 mm

Varie:

- Porta led 5 mm da pannello (3 pz.)
- flat cable 15 cm 8 vie
- flat cable 17 cm 2 vie
- Strip femmina 8 poli
- Strip femmina 1 polo (4 pz.)
- Strip femmina 4 poli
- Strip femmina 2 poli (4 pz.)
- Strip maschio 6 poli
- Distanziale M/F 15 mm (4 pz.)
- Dado 3 MA (8 pz.)
- Vite 20 mm 3 MA (4 pz.)
- Circuito stampato



delle letture di potenza istantanea. In comandi sono i seguenti:

- **"1.Potenza MAX"**: indicazione del valore massimo tra tutte le letture di potenza fatte. Accendendo a questo comando verrà chiesto di premere UP/DOWN, rispettivamente per il 'Reset' del valore attuale (così da iniziare un nuovo periodo di monitoraggio), e per la lettura ('Leggi') del valore fino a quel momento registrato.

- **"2.Potenza Media"**: calcolo della potenza media entro un periodo di 1 minuto. Al termine viene visualizzata la potenza media.

- **"3.Ins.Pot.Mas."**: inserimento della massima potenza consentita in uscita prima dell'intervento limitativo del sistema. Per modificare il valore (da un minimo di 400W ad un massimo di 6.000 W), basta premere ripetutamente o mantenere premuto UP (per aumentare) o DOWN (per diminuire) e poi confermare con il tasto SET.

L'incremento o decremento avviene a passi di 50 W.

- **"4.Priorita' MAX"**: inserimento del massimo livello di priorità dei ricevitori. Il comando prevede l'inserimento di un nuovo valore (minimo di 1 e massimo di 6) premendo UP, e la lettura del valore attuale premendo DOWN. La modifica si effettua premendo UP ripetutamente fino al valore desiderato, poi si conferma con SET.

- **"5.Forza ON liv."**: consente l'attivazione forzata di tutti gli attuatori, quindi degli apparecchi elettrici ad essi collegati. Dapprima vengono attivati istantaneamente gli Smart-Rx tramite un codice speciale, e a seguire, tutti i ricevitori Motorola secondo il loro livello di priorità (a condizione che il jumper di configurazione RXT sia chiuso).

- **"6.Forza OFF liv"**: consente la disattivazione forzata di tutti gli attuatori. Dapprima vengono disattivati istantaneamente

gli Smart-Rx tramite un codice speciale, e a seguire tutti i ricevitori Motorola secondo il loro livello di priorità (a condizione che il jumper RXT sia chiuso).

- **"7.NuovoCod.Base"**: consente la modifica del codice "Base" Motorola (6 bit MSB) emesso dalla centralina. Premendo UP si potrà personalizzare bit per bit tale codice "Base" (Zero, One, Open) entro un massimo di 729 possibili combinazioni (SET per confermare). Premendo DOWN, invece, sarà possibile modificare il livello priorità emesso subito dopo il codice base ogni volta che si esegue il comando "12.Taratura RF". La procedura viene conclusa con la modifica automatica del codice base degli Smart-RX (quelli tradizionali vanno codificati manualmente).

- **"8.Vis. Cod.Base"**: consente la visualizzazione del codice base (Zero, One, Open);

- **"9.Vis.liv.prior"**: permette la visualizzazione dei livelli di

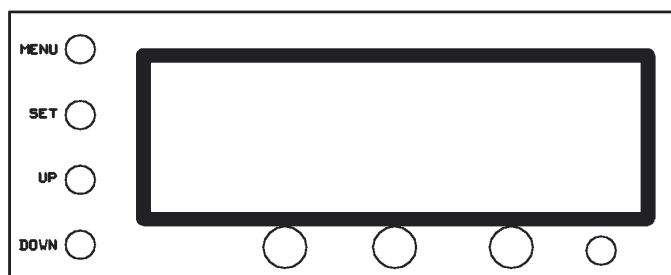
priorità degli Smart-RX direttamente (e su ciascuno di essi) tramite il loro led verde: serie di lampeggi veloci = priorità 0, altrimenti 'n' lampeggi = priorità 'n'. Questa visualizzazione si ripete per cinque volte con cadenza di 4 secondi.

- "10. **BuzzerON/OFF**":

consente l'attivazione/disattivazione del cicalino per la segnalazione del sovraccarico. Bisogna commutare lo stato in ON/OFF tramite il pulsante UP e poi confermare con SET.

- "11. **ResetSistema**": consente il reset del sistema con il ripristino dei valori predefiniti. Bisogna tenere premuti contemporaneamente UP e DOWN per 5 secondi durante i quali il led verde lampeggerà ogni 0,5 secondi. In caso contrario la sequenza di reset non verrà eseguita. I valori predefiniti sono:

- Codice Base: **H H H H H H**;
 - Livello priorità per la codifica degli Rx: **0**;
 - Buzzer: **ON** ;
 - Power MAX: **6.000 W** ;
 - Livello Priorità massima: **6**
- "12. **Taratura RF** ": consente



Mascherina del pannello di comando avente dimensioni di 100 x 41 mm.

l'emissione continua di un codice Motorola costituito dal codice base e dal codice di priorità definito col comando 7. Consente la codifica degli Smart-Rx (autoapprendimento del codice base e del livello di priorità) e per il test di ricezione del segnale RF. La procedura viene terminata tenendo premuto SET fino allo spegnimento per circa un secondo del led giallo; subito dopo viene emesso per pochi istanti un secondo codice in grado di disattivare quei ricevitori attivati in precedenza e aventi lo stesso livello di priorità. Una volta codificato uno Smart-Rx (o di un trasmettitore Motorola), lo si può installare collegandolo ad una presa dell'impianto elettrico controllato dal sistema Antiblackout. Questo comando, inoltre, consente la taratura della frequenza di clock del codice Motorola, che può essere aumentata modificata agendo su UP e DOWN. Tale regolazione non è sempre necessaria, ma può essere utile per mantenere la compatibilità di funzionamento con tutti i ricevitori Motorola classici, ovvero quelli senza il PIC-decoder.

CONFIGURAZIONE INIZIALE

Il sistema necessita di alcune impostazioni iniziali affinché intervenga al momento giusto. La sequenza consigliata è la seguente:

1. **Inserimento della potenza massima consentita:** utilizzando il comando 3 inserite la soglia di potenza raggiunta

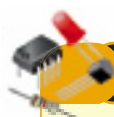
la quale il sistema interverrà nel controllo del sovraccarico, disattivando gli attuatori di uno o più livelli di priorità. Se, ad esempio, l'utenza elettrica ha un limite di potenza pari a 3 kW, questo valore può essere il limite per la centralina. Il valore predefinito è 6 kW.

2. **Inserimento del massimo livello di priorità:** utilizzando il comando 4 inserite il massimo livello priorità (da 1 a 6) gestito dall'Antiblackout. I livelli, al massimo, sono 7, ma tale numero può essere ridotto in base alle proprie esigenze. Ciò per evitare che il sistema perda tempo a disattivare livelli non utilizzati. Il valore predefinito è 6.

3. **Inserimento del Codice Base:** tutti i codici di comando emessi dalla centralina sono di tipo Motorola con codifica three-state a 9 bit: i primi sei sono riservati al Codice Base, ovvero la parola di identificazione del sistema Antiblackout. Per modificarlo utilizzare il comando 7. Il valore predefinito è H H H H H H.

4. **Attivazione cicalino:** utilizzando il comando 10 è possibile attivare o disattivare il cicalino di segnalazione. Se ON (valore predefinito) il cicalino segnala la condizione di sovraccarico quando la disattivazione di tutti i livelli priorità non ha portato il consumo elettrico al di sotto della potenza massima.

Appuntamento al prossimo numero della rivista, con la descrizione dei ricevitori/attuatori. ■



per il MATERIALE

Tutti i componenti utilizzati in questo progetto sono facilmente reperibili in commercio. Il firmware ed il master dello stampato possono essere scaricati dal sito della rivista (www.elettronica.in.it). L'integrato MCP3905A costa 4,20 Euro mentre il microcontrollore vergine PIC16F876-I/SP costa 8 Euro. Quest'ultimo va programmato direttamente sulla basetta tramite un programmatore munito di connettore ICSP. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:

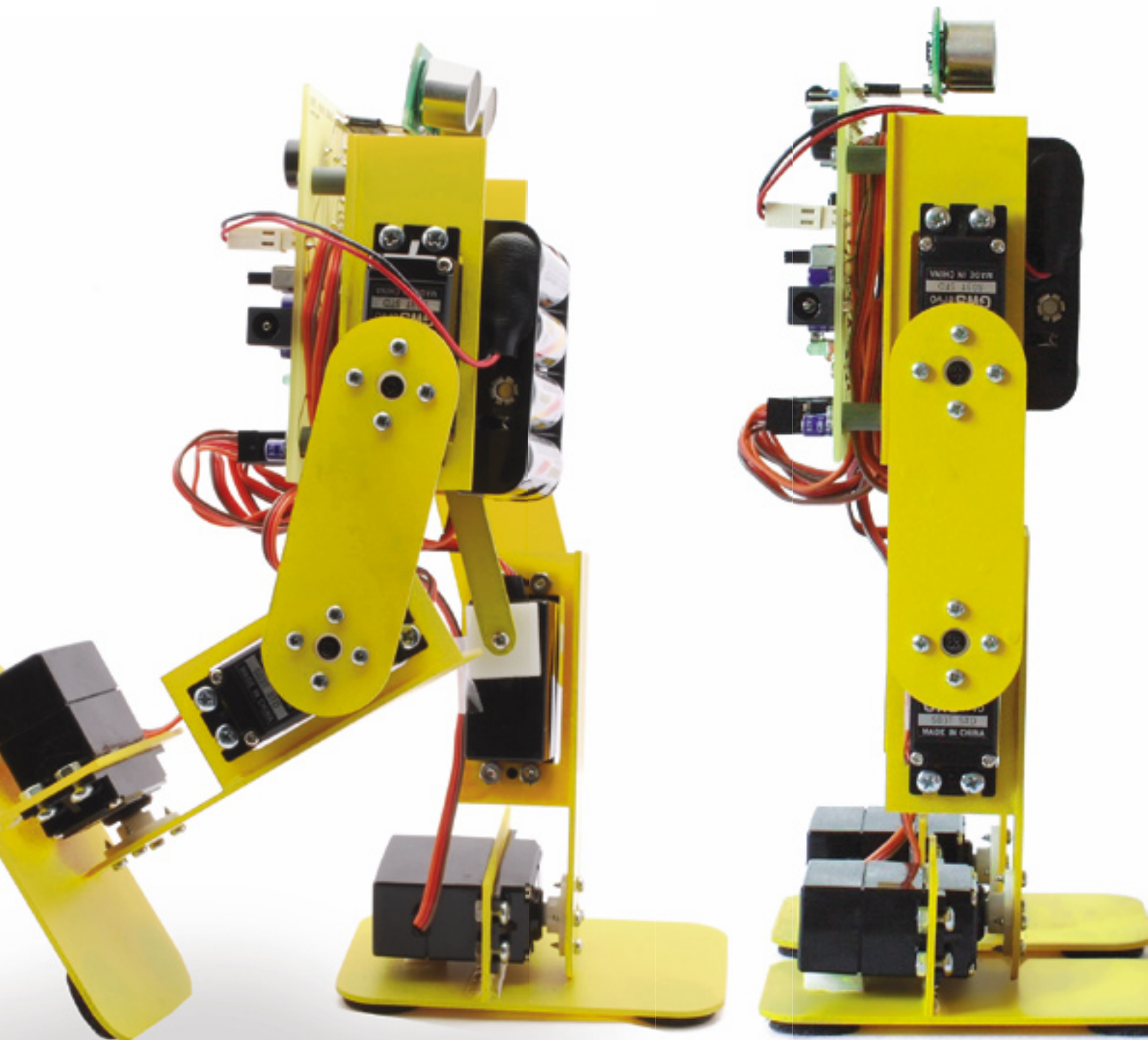
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)

Tel: 0331-799775 • Fax: 0331-792287
<http://www.futurashop.it>

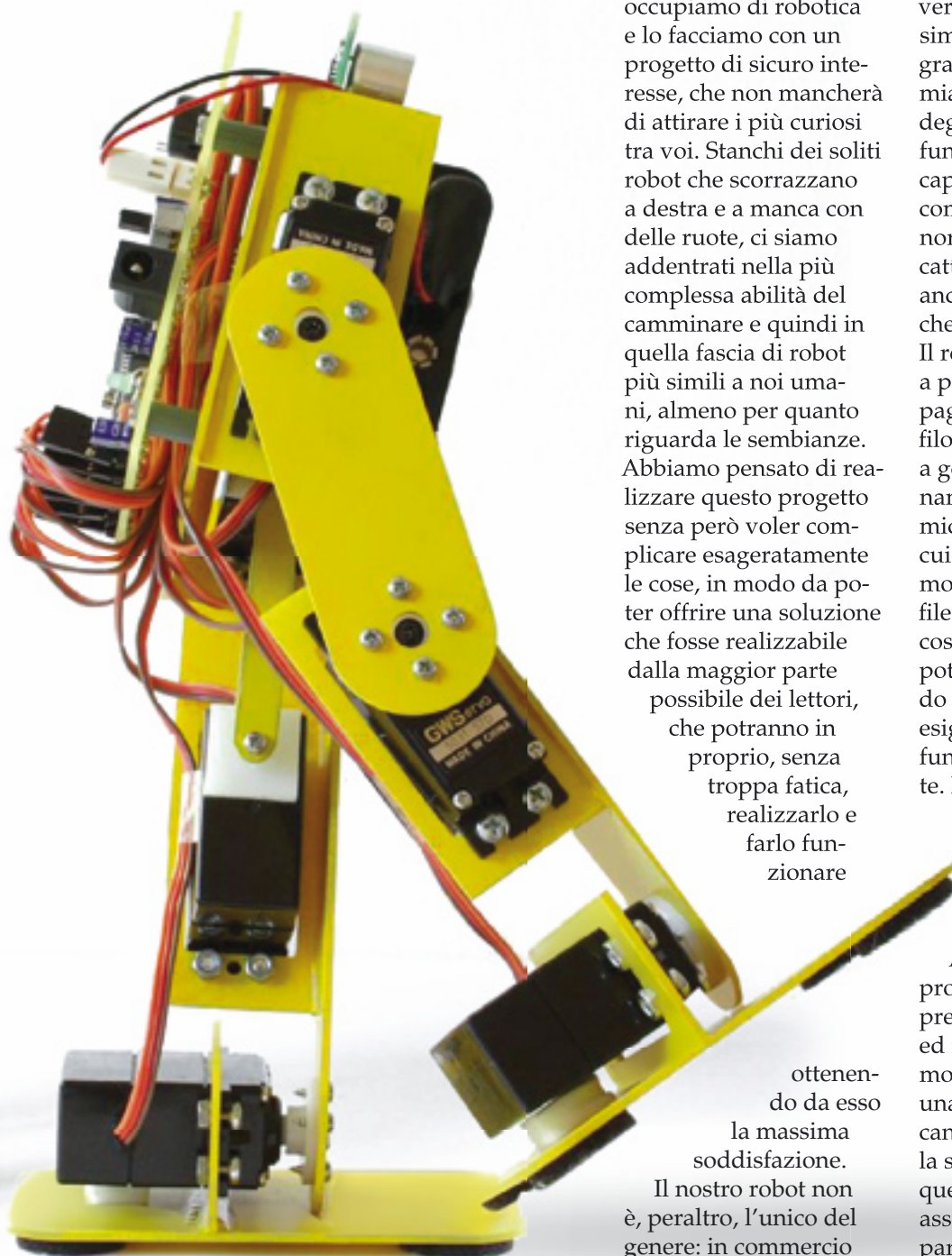
ROBOT BIPEDE

a controllo remoto

di MIRCO SEGATELLO



Cammina, si piega, può addirittura calciare una pallina da tennis.
È un giocattolo o qualcosa di più? Scopritelo leggendo queste pagine...



Ancora una volta ci occupiamo di robotica e lo facciamo con un progetto di sicuro interesse, che non mancherà di attirare i più curiosi tra voi. Stanchi dei soliti robot che scorrazzano a destra e a manca con delle ruote, ci siamo addentrati nella più complessa abilità del camminare e quindi in quella fascia di robot più simili a noi umani, almeno per quanto riguarda le sembianze. Abbiamo pensato di realizzare questo progetto senza però voler complicare esageratamente le cose, in modo da poter offrire una soluzione che fosse realizzabile dalla maggior parte possibile dei lettori, che potranno in proprio, senza troppa fatica, realizzarlo e farlo funzionare

ottenendo da esso la massima soddisfazione. Il nostro robot non è, peraltro, l'unico del genere: in commercio

sono disponibili diversi giocattoli evoluti simili, che ricalcano a grandi linee la fisionomia e le abilità motorie degli esseri umani; le funzioni di cui sono capaci a volte sono così complesse da sfiorare la normale idea di giocattolo ed interessare anche i papà, piuttosto che i figli.

Il robot che andiamo a presentare in queste pagine si basa su una filosofia open-source: a gestire il suo funzionamento provvede un microcontrollore, del cui software rendiamo disponibili tutti i file in codice sorgente, cosicché chi se la sentirà potrà modificare secondo il proprio gusto o esigenze specifiche le funzioni implementate. Nulla impedirà di

realizzare nuovi movimenti, anche complessi, pensati a proprio piacimento.

Anche in questo progetto, oltre alla parte prettamente elettronica ed informatica, dobbiamo disporre anche di una componente meccanica necessaria per la struttura del robot; questo aspetto non deve assolutamente preoccupare il lettore, in quanto

La meccanica

il tutto è stato risolto attraverso l'impiego di circuiti stampati; in altre parole, lo chassis del robot si compone di pezzi di vetronite ramata da assemblare saldandoli a stagno. Oltre alle foto e all'articolo disponibili in queste pagine, sul sito della rivista (www.elettronica.in.it) accedendo al numero di questo fascicolo (n° 133) e facendo clic sull'apposito link, è disponibile un filmato che consente di vedere in pratica e in azione le potenzialità del nostro piccolo ma grande automa.

PROBLEMATICHE DI PROGETTO

Prima di descrivere in dettaglio la costruzione, facciamo una breve introduzione che vi permetterà di capire le scelte da noi fatte e in quale modo abbiamo condotto lo sviluppo di questo progetto. Partiamo da questa premessa: se vogliamo che un robot cammini in modo simile a noi umani, dobbiamo poter riprodurre determinati tipi di movimenti: in questo caso quelli delle gambe. Per avere un'idea di quanto ciò sia complesso, provate a fare alcuni passi osservando quali movimenti state eseguendo e soffermatevi ad analizzare quanti muscoli ed articolazioni avete utilizzato: rimarrete stupefatti dalla loro complessità e dalla capacità del nostro cervello di un così sofisticato coordinamento. Non per niente la postura eretta e il camminare con due gambe sono stati un passo molto importante dell'evoluzione umana e senza andare tanto lontano, basti pensare ad un bambino che compie i suoi primi passi: per niente facile! Dal bacino partono i due femori che permettono alla gamba di muoversi in avanti, indietro, lateralmente e di ruotare, poi abbiamo il ginocchio, che flette la seconda parte della gamba e poi arriviamo al piede con il suo

movimento, anche se di pochi gradi, in tutte le direzioni; infine, pensate alla mobilità delle dita del piede. Tutti questi movimenti sono coordinati alla perfezione, permettendo il moto in tutte le direzioni anche nei percorsi sconnessi. Nel campo della robotica, riprodurre una simile flessibilità, un apparato motorio simile a quello umano, è considerato compito non da poco; attualmente un buon esempio è

rappresentato dal robot ASIMO (il cui progetto è stato finanziato dalla giapponese HONDA) che rappresenta il miglior esempio di automa umanoide. Potete facilmente trovare alcuni video su Internet, dove vengono illustrate le sue capacità. Il nostro progetto, molto più semplice, parte dall'idea di poter ottenere l'abilità motoria con due arti, utilizzando il minor numero possibile di giunzioni; in questo

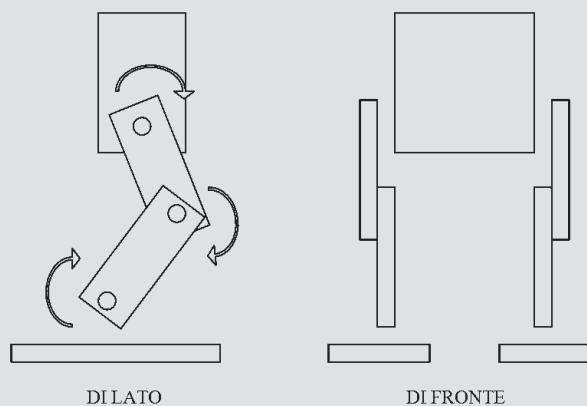


Fig. 1 - Movimenti dell'articolazione della gamba del nostro robot.

Fig. 3 - Assemblaggio della parte inferiore della gamba.

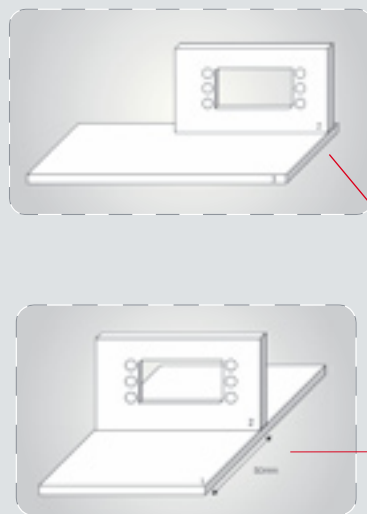


Fig. 2 - Assemblaggio delle parti che compongono un piede.

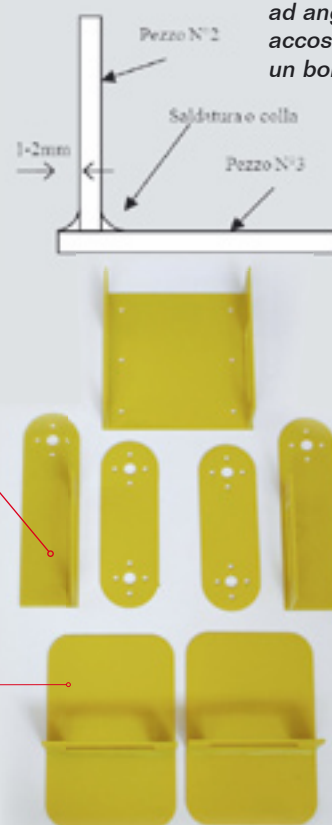
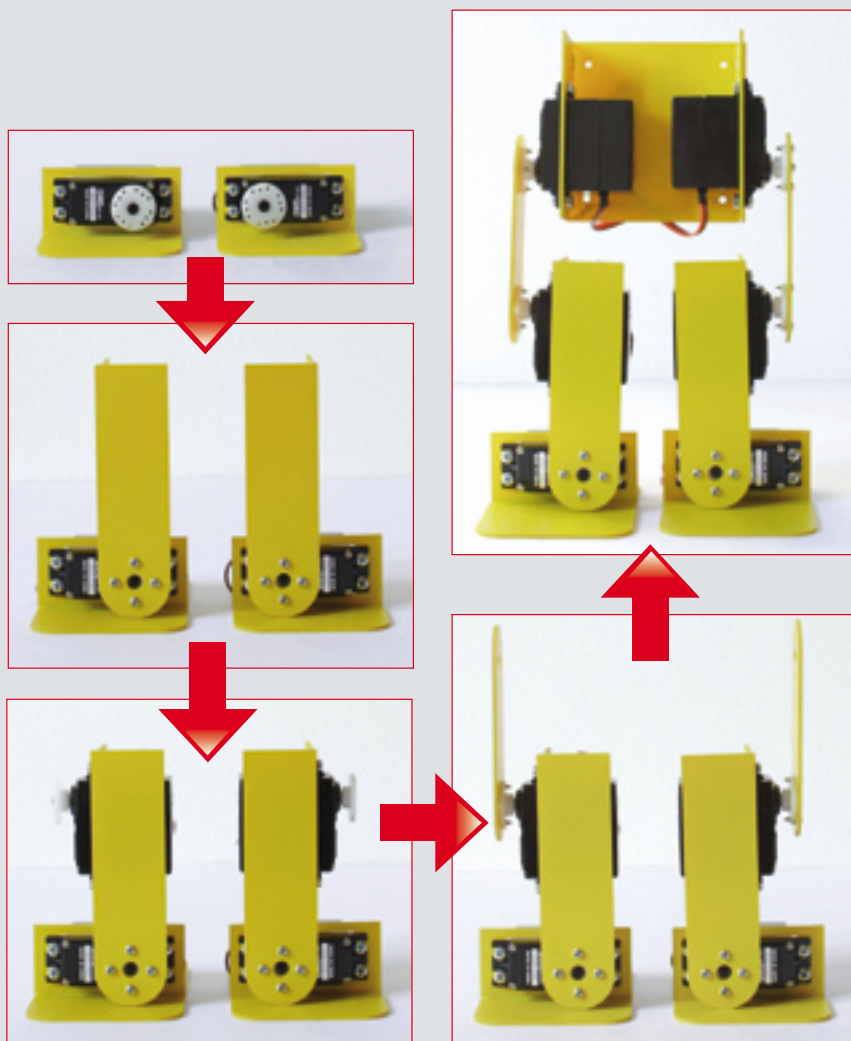


Fig. 4 - Le parti ad angolo vanno accostate lasciando un bordo di 2 mm.

Fig. 5 - Insieme delle parti preassemblate.



Tutte le fasi di assemblaggio delle gambe e del corpo del robot; mancano solamente i circuiti elettronici (scheda di controllo e modulo sonar) e il vano delle batterie, oltre ai collegamenti. Dopo aver realizzato i piedi, inseritevi i servocomandi e avvitate il mozzo di questi ultimi sulle parti inferiori delle gambe. In cima a questi montate i servo corrispondenti e poi, sui rispettivi mozzi, avvitate le parti superiori. Queste ultime dovete poi fissarle superiormente ai servocomandi che, preventivamente, avrete già introdotto nel corpo del robot. Il risultato lo vedete nella foto in alto a destra. Solo allora potete fissare le colonnine per montare la scheda di controllo.

modo possiamo ridurre sia i costi che la complessità meccanica, condizioni che, per un primo approccio alla materia *robot*, risultano obiettivi primari. Accettando alcuni limiti e qualche compromesso, con sole tre articolazioni per gamba è possibile costruire un robot in grado di muoversi in modo molto simile a noi umani. I movimenti essenziali sono l'articolazione dell'anca, quella del ginocchio e quella del

piede, che ovviamente avverrà in una sola direzione (Fig. 1). Scelti i movimenti, è stato essenziale decidere le dimensioni finali del robot, il peso, il tipo di azionamento e quali potenze erano in gioco. Altra condizione importante era che il robot potesse risultare facile da realizzare anche a chi ha conoscenze e disponibilità limitate; ecco perché per realizzare gli attuatori abbiamo optato per l'utilizzo di

servocomandi da modellismo. Questi componenti li abbiamo già usati in precedenti progetti e li conosciamo molto bene: sono facili da reperire e il loro costo non è eccessivo; ma, cosa fondamentale, oltre alla parte di potenza dispongono al loro interno anche della logica di comando, il che semplifica non poco la circuizione logica che dovrà comandare i vari movimenti. Nel numero 116 di *ElettronicaIn*, nell'articolo riguardante il braccio meccanico, trovate una spiegazione più approfondita di questi servo da modellismo, in special modo per quel che concerne il loro funzionamento.

LA MECCANICA

Tra i vari servocomandi disponibili sul mercato, abbiamo optato per la versione standard, che ha un lunghezza tipica di 40 mm e un peso di circa 40÷50 g; da questo siamo partiti per il progetto della parte meccanica che avrebbe realizzato il robot. Le dimensioni e il peso dei servo hanno, a questo punto, dettato la dimensione finale della struttura, dalla quale siamo partiti con il disegno dei pezzi meccanici idonei a realizzarla. Scartate tutte le soluzioni riguardanti plastiche ed alluminio, che devono essere tagliati e sagomati, abbiamo preferito utilizzare quello che è il nostro pane quotidiano: i circuiti stampati. La vetronite di cui sono fatti è un materiale eccezionale per robustezza e facilità di lavorazione; inoltre, la ramatura della superficie permette facili saldature. Le parti meccaniche componenti il nostro robot sono ottenute assemblando dei pezzetti di vetronite ramata, i cui modelli sono riportati in queste pagine. Nel disegno delle parti meccaniche abbiamo privilegiato la semplicità realizzativa, in

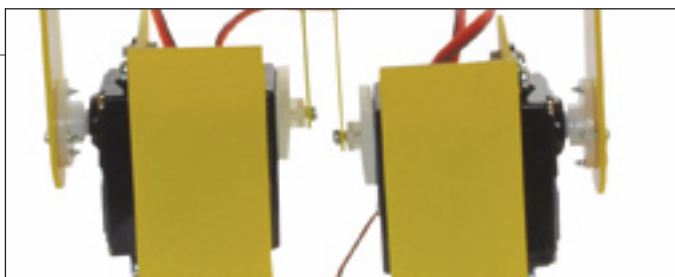


Fig. 6 - Dettaglio di montaggio relativo ai pezzi N°7.

modo che il lavoro di sagomatura potesse essere realizzato anche da ditte specializzate in PCB alla stregua di un comune circuito per elettronica; sono stati previsti anche tutti i fori necessari, comprese le scanalature per i servocomandi. Il disegno delle basette ramate può essere altresì scaricato dalla sezione *Download* del sito www.elettronica.in.it, sotto forma di file *gerber* (apribile con *GC-preview*, ad esempio) oppure *tiff* (*bipe.tif*) stampabile in dimensioni reali su foglio A4. Tutti i pezzi necessari possono essere ricavati da una lastra di vetronite ramata da entrambe le facce, delle dimensioni di un foglio A4. Per evitare questa gravosa fase di lavoro, potete scaricare dal nostro sito Internet i file relativi ai master dei circuiti stampati occorrenti (scheda di controllo e telecomando) e quelli per le parti meccaniche e rivolgervi poi a una ditta specializzata nella realizzazione di PCB come la MDSRL (www.mdsrl.it) la quale vi fornirà tutti i pezzi già tagliati e forati con estrema precisione. Risolto il problema della struttura meccanica, procuratevi sei servocomandi da modellismo standard; per il nostro robot sono stati usati i GWS S03T, disponibili presso la ditta Futura Elettronica di via Adige 11, Gallarate (VA). Chi volesse, potrà realizzare i pezzi meccanici anche con della semplice vetronite non ramata, oppure con del plexiglass dello spessore di 2 mm, fermo restando che in questo caso per unirli dovrà utilizzare della colla ade-

guata. Nel caso della vetronite sarà d'obbligo l'uso di colla a base epossidica bicomponente, quella che in commercio si trova in una doppia siringa e che va mischiata per poter indurire.

Per completare il lavoro, non vi resta che reperire il seguente materiale facilmente acquistabile in negozi di elettronica, ferramenta o centri per il fai da te:

- N° 24 viti 3x6 mm;
- N° 4 viti 3x12 mm;
- N° 28 dadi filettati 3 mm;
- N° 28 viti autofilettanti;
- N° 4 supporti adesivi altezza 6 mm;
- N° 4 distanziatori in plastica.

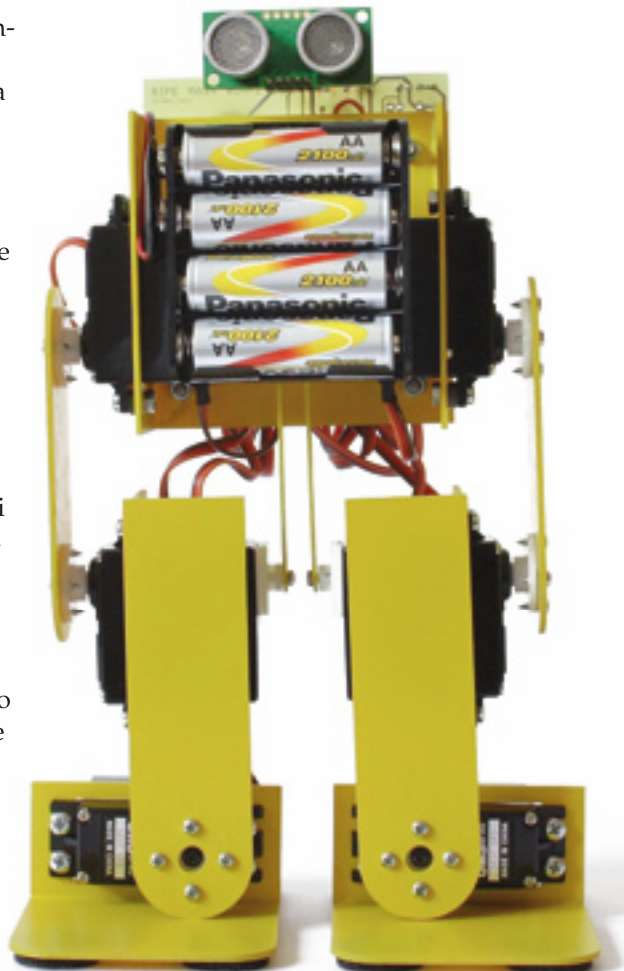
Procedete ora all'assemblaggio, nel seguente ordine: sul piede, pezzo N° 1, saldate trasversalmente a 90° il pezzo N° 2, mantenendolo a 50 mm dal bordo anteriore (Fig. 2). Fate la stessa cosa per l'altro piede.

Sul pezzo N°3 saldate a 90° il pezzo N° 2. Fate la stessa cosa con l'altro pezzo N° 3 e N° 2; assemblate così una parte delle gambe (Fig. 3).

Saldate adesso i due pezzi N° 5 sul corpo N° 6. Quando unite due pezzi, fate in modo da lasciare dal bordo uno spazio di circa 1÷2 mm, in modo da permettere la saldatura (o l'incollaggio) su entrambi i lati (Fig. 4); senza questo margine, sarà possibile colare lo stagno da una sola parte della giuntura, il che renderà il pezzo piuttosto debole e facilmente deformabile. A lavoro ultimato ottenete le parti preassemblate visibili nella Fig. 5, che formano le gambe del robot. A questo punto potete iniziare

l'assemblaggio dei servocomandi, procedendo secondo la sequenza illustrata nelle figure dell'apposito riquadro.

I due pezzi N° 7 rappresentano l'interno coscia e sono necessari perché durante l'inclinazione laterale del robot il perno del servo, essendo leggermente lasco, produce una certa flessione. Pur non essendo strettamente necessario, se ne consiglia l'utilizzo a meno di non usare servo dotati di cuscinetti, il cui perno non presenta particolare gioco. Utilizzate i quattro supporti adesivi e tagliatene il perno, quindi fissateli sui due servocomandi relativi all'anca ed al ginocchio sul lato opposto a quello del perno, ma in asse con esso. I pezzi N° 7 vanno fissati sui due supporti adesivi



Vista anteriore del robot bipede completo: il davanti è quello dove si affacciano le capsule a ultrasuoni del sonar.

Il robot "vede" grazie ad un sonar a 40 kHz che gli permette di rilevare gli ostacoli.



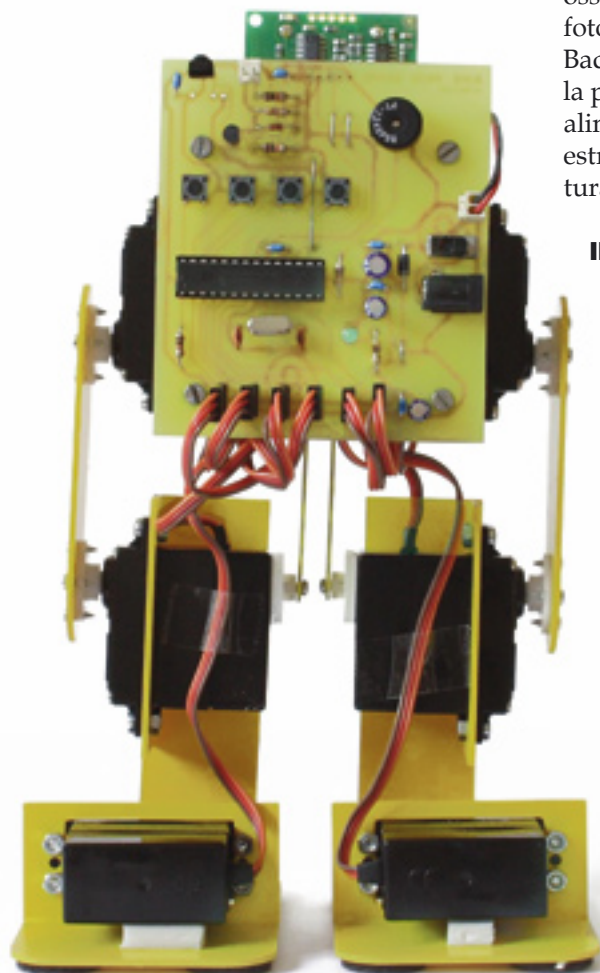
tramite viti e devono essere liberi di ruotare; dovete forarli con una punta esattamente dello stesso diametro della vite usata per fissarli, in modo da non lasciare alcun gioco. È importante che le giunzioni del robot relative ad anca e ginocchio risultino a 180° tra loro, mentre il piede stia a 90°; semmai fate in modo che il busto venga leggermente inclinato in avanti. Tutto questo è valido se i servo si trovano in posizione di neutro come di solito accade quando sono nuovi; se non ne avete la certezza, è bene per prima cosa completare la scheda di controllo ed utilizzarla per riportare i servocomandi nella posizione di zero. Notate che non è necessario che siano rispettate con assoluta precisione tutte le

inclinazioni, perché la scheda di controllo permette un setup della posizione di neutro dei servo utile per correggere le piccole imperfezioni meccaniche. Per ultimo fissate le batterie, quattro pile stilo ricaricabili (nell'esempio mostrato) che è previsto prendano posto in un portapile piatto (in linea), quindi fissate con del biadesivo (o del velcro adesivo) sulla parte anteriore del robot. Per ultimo applicate dei feltrini adesivi sotto i piedi. Con i servocomandi e le viti indicati non sono necessari fori di alcun tipo ed il fissaggio dei pezzi alla squadretta dei servo avviene molto velocemente tramite quattro viti. Nel caso abbiate dubbi, osservate con attenzione tutte le foto riportate in questo articolo. Badate che è possibile muovere la posizione di un servo non alimentato, a patto di farlo con estrema delicatezza, pena la rottura degli ingranaggi interni.

IL CIRCUITO DI COMANDO

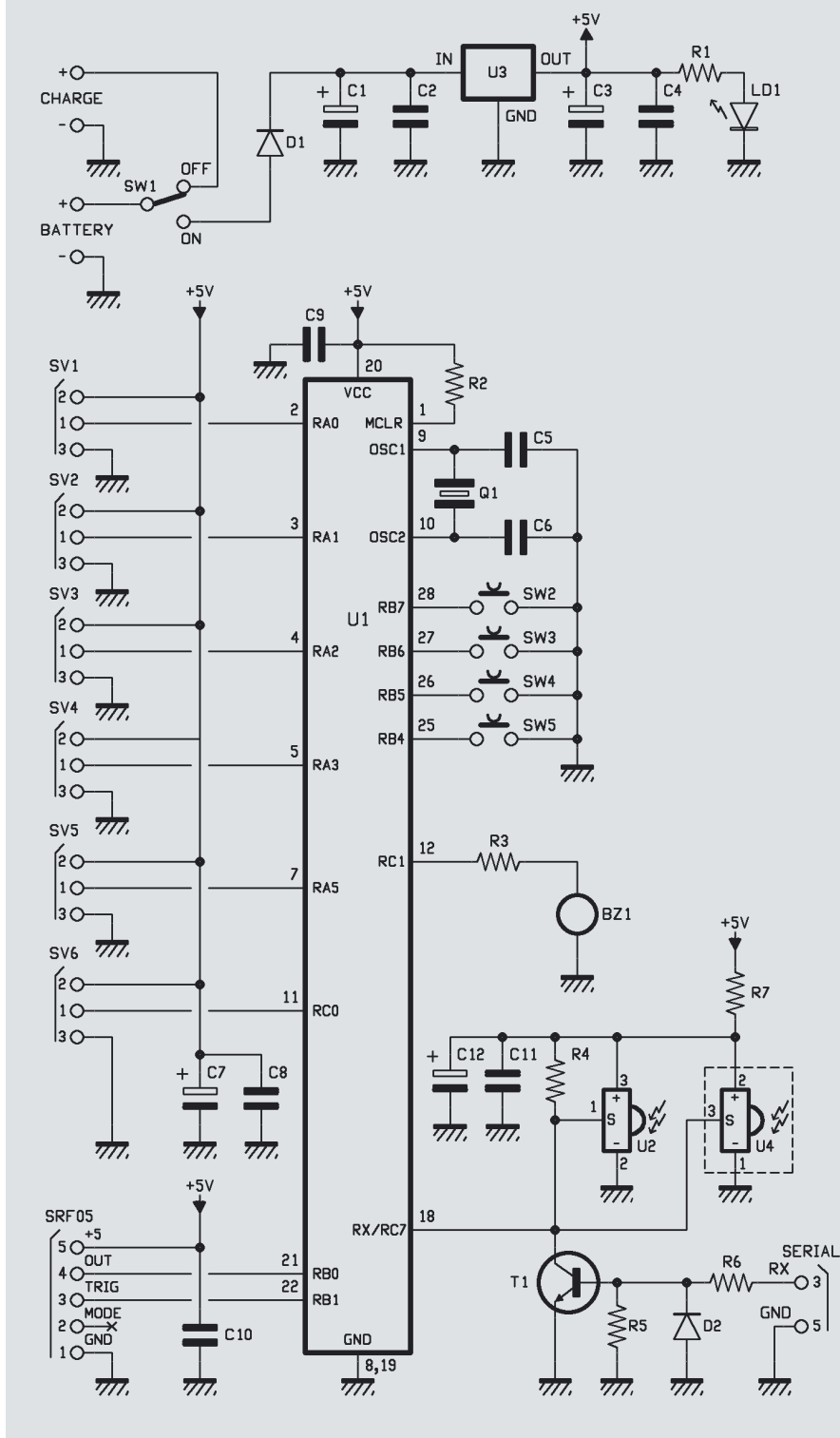
La scheda di comando è equipaggiata con un PIC16F876, il quale, opportunamente programmato, gestisce la totalità delle funzioni, compresa l'attività dei sei servocomandi, coordinandone il movimento e quindi presiedendo alla movimentazione generale del robot. Il circuito di alimentazione è molto flessibile e si presta a svariate soluzioni; esso fa capo connettore J1. Le possibili soluzioni sono le seguenti: alimentazione con 6 celle (ricaricabili o meno); in questo caso sono neces-

sari tutti i componenti indicati nello schema. Semmai potete eliminare D1, che viene usato solo per impedire danneggiamenti alla scheda nel caso di inversione di polarità nell'alimentazione; siccome crea una caduta di tensione di 0,7 volt, se siete sicuri della polarità di alimentazione potete ometterlo ed al suo posto saldare un ponticello fatto con un filo di rame. Questa soluzione risulta molto interessante se si alimenta il robot con una batteria a due celle Li-Po che fornisce una tensione nominale di 7,4 V. Un'altra soluzione è l'alimentazione a 4 batterie (ricaricabili o meno); in questo caso dovete omettere lo stabilizzatore di tensione U3, che risulta inutile, e sostituirlo con un ponticello fatto con un filo di rame che cortocircuita il pin 1 con il 3. Tale soluzione, peraltro adottata nel prototipo che vedete nelle fotografie in queste pagine, consente di usare un minor numero di batterie e quindi avere un peso minore. La presenza del diodo D1 è fondamentale perché, grazie alla sua caduta di tensione, anche quando le batterie sono completamente cariche la tensione nel circuito non supererà i 5,5 volt, limite oltre il quale il PIC potrebbe danneggiarsi. Nessun problema, invece, per i servocomandi, i quali hanno una tensione massima di funzionamento di 6 V. Consigliamo da subito l'utilizzo di batterie ricaricabili, meglio se di elevata capacità, in quanto sei servo da modellismo che funzionano contemporaneamente



Vista posteriore del robot completo: notate come i servocomandi sono fissati ai piedi e alle gambe.

[schema **SCHEDA ROBOT**]



assorbono parecchia corrente e le normali pile a secco non avrebbero lunga vita. Nell'esempio presentato abbiamo usato quattro batterie ricaricabili formato AA

da 2100 mA/h. Il deviatore SW1 fa da interruttore di accensione e, se posto in posizione OFF, mette in comunicazione il pacco batterie con il connettore J9, utile

per collegare un caricabatteria esterno e poter così eseguire la ricarica del pacco batterie senza staccare alcuna delle connessioni previste per il normale funzionamento. Con una piccola modifica allo stampato, J9 potrebbe essere usato per alimentare esternamente il robot tramite un piccolo alimentatore stabilizzato, in modo da permettere un funzionamento ininterrotto.

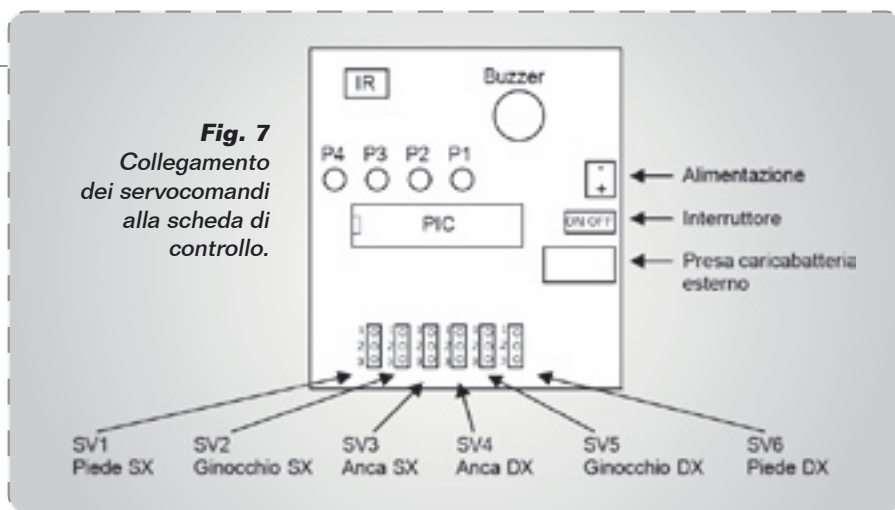
Il led D2 indica, accendendosi a luce piena, quando la scheda di controllo è correttamente alimentata; se emette una luce fioca, vuol dire che le batterie sono scariche. La scheda prevede quattro pulsanti che servono ad attivare altrettante funzioni base; è però previsto di impartire comandi anche dall'esterno, che giungono al pin RC7 del PIC tramite un telecomando senza fili o un PC connesso via cavo. Il circuito relativo a Q1, R5, R6, D3 fa capo al connettore J8 ed è idoneo ad accettare i segnali provenienti dalla seriale del PC. In parallelo alla stessa RC7, troviamo i componenti siglati U2 e U4: si tratta di sensori all'infrarosso idonei a ricevere i comandi da un telecomando IR e inviare gli impulsi corrispondenti sulla linea comune a quella del converter seriale che serve alla connessione con il computer. Il fatto che il sensore infrarosso lavori in parallelo con il segnale dalla seriale non deve impensierire, in quanto nell'uso del robot si sceglie una sola modalità di comando (non avrebbe senso inviare contemporaneamente il segnale dal telecomando IR e dal computer); la connessione realizzata corrisponde a una porta logica OR con ingressi open-collector. Si noti che nello schema figurano due integrati, ma non perché bisogna montarli entrambi simultaneamente: i due vanno usati uno in alternativa

all'altro; insomma, la scheda di controllo è in grado di accettare l'installazione di entrambi i tipi, ma effettivamente dovrà essere montato solo uno dei due. Abbiamo previsto la possibilità di montare due diversi moduli ricevitori ad infrarossi per ottenere un circuito molto flessibile, in grado di lavorare con i componenti che più vi aggradano e che più facilmente riuscite a reperire. Oltre ai due elementi previsti, un altro ricevitore idoneo potrebbe essere il sensore SFH506 prodotto dalla Siemens. Se pensate di utilizzare altre versioni, rammentate che debbono funzionare con una tensione di 5 volt e che sia adatto alla ricezione di segnali a 38 Hz. Proseguendo con l'esame dello schema elettrico, notiamo che nel connettore J11 è previsto l'inserimento di un sensore tipo radar ad ultrasuoni, modello SRF05 (è distribuito dalla ditta Futura Elettronica) il cui scopo è quello di rilevare la prossimità di eventuali ostacoli che dovessero presentarsi di fronte al robot quando questo cammina in avanti; il segnale uscente dal radar ultrasonico dà un'indicazione della presenza di oggetti nel campo che il dispositivo copre e sulla distanza da essi, e viene letto dal microcontrollore il cui firmware provvede a comandare i servo in modo che il robot cambi direzione nel caso vi siano ostacoli ravvicinati. In ultimo, notate i sei connettori da J2 a J7, che servono a connettere altrettanti servocomandi. Il buzzer, come vedremo, sarà utile quale monitor dello stato di funzionamento del robot.



Corrispondenza tra i colori del cavo di collegamento e i riferimenti del c.s.

Fig. 7
Collegamento
dei servocomandi
alla scheda di
controllo.



REALIZZAZIONE DELLA SCHEDA DI CONTROLLO

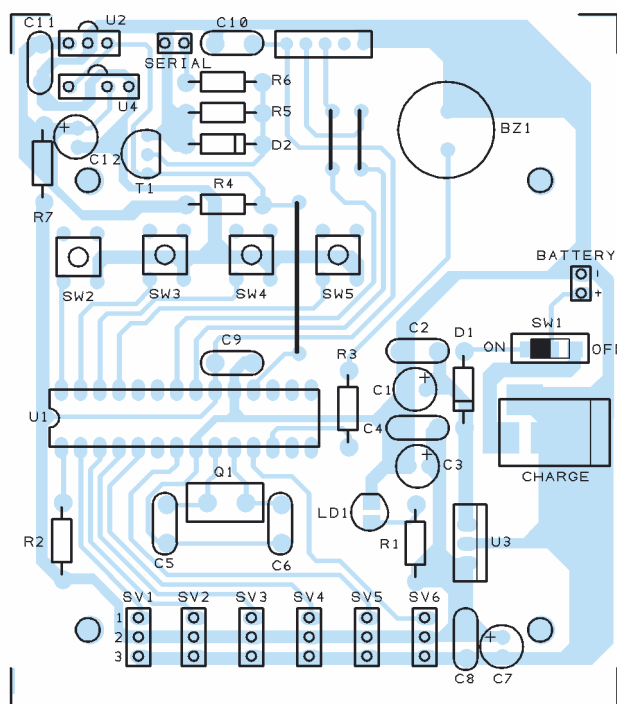
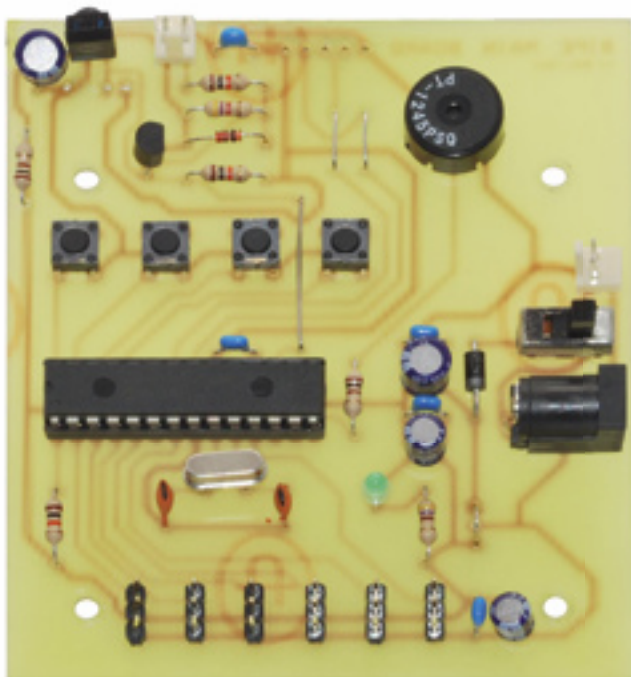
La scheda di comando è semplice da costruire e non presenta particolari problematiche; per prepararla iniziate, come al solito, realizzando il circuito stampato monofaccia, a partire dai file contenenti le tracce che trovate nel sito della rivista. Le piste di dimensioni generose e la distanza tra i componenti facilitano la realizzazione. Iniziate il montaggio partendo con i componenti a più basso profilo, come resistenze, diodi e zoccoli; a seguire, pulsanti, condensatori e quindi tutto il resto. Per ultimo saldate il sensore IR (uno o l'altro secondo disponibilità) in verticale in modo che "guardi" verso l'alto. Nel nostro esempio abbiamo utilizzato il sensore PNA4602, il quale, essendo leggermente più sensibile di altri simili, permette una maggiore portata del telecomando. Ad assemblaggio ultimato, la scheda va fissata sul retro del robot con quattro viti, utilizzando altrettanti distanziatori in modo da lasciare uno spazio di 5÷10 mm tra essa e il corpo del robot. Utilizzate questo spazio per inserire i cavi dei servocomandi, che altrimenti penderebbero all'esterno; è

importante che i servo vengano collegati come indicato nell'apposita figura.

COLLAUDO E TARATURA

Terminato l'assemblaggio della meccanica e dell'elettronica, assicuratevi che sia tutto installato correttamente; per il momento non inserite il PIC e non connettete i servocomandi, ma limitatevi a fornire tensione e controllare che il led si accenda. Procurato un tester, controllate che la tensione di alimentazione sia quella corretta; a questo punto inserite il PIC programmato e tutti i servo ed accendete la scheda. Notate come tutti i servo vengono posizionati automaticamente nella posizione di neutro. In questa situazione, il robot, dovrà risultare nella sua postura definitiva o quasi: eretto, in piedi e abbastanza dritto; se notate che qualche giunzione è troppo sfalsata rispetto a come dovrebbe essere, potete eseguire una regolazione grossolana modificando la posizione della squadretta del servo interessato, svitando la vite del perno e riposizionando la squadretta stessa. Ogni servo, inoltre, può essere spostato di un paio di millimetri all'interno della sua sede, cosa che ci permette di compensare eventuali imprecisioni nelle saldature. Giunti a questo punto, è bene procedere alla taratura "di fino" della posizione dei servocomandi; per compierla spegnete

[piano di **MONTAGGIO** della **SCHEDA ROBOT**]



Elenco Componenti:

R1: 470 ohm
R2, R4, R6: 10 kohm
R3, R7: 100 ohm
R5: 4,7 kohm
D1: 1N4007
D2: 1N4148
LD1: led verde 3 mm
C1: 100 μ F 16 VL elettrolitico
C2, C4: 100 nF multistrato
C3, C7: 10 μ F 16 VL elettrolitico
C5, C6: 22 pF ceramico

C8÷C11: 100 nF multistrato
C12: 100 μ F 16 VL elettrolitico
SRF05: Sensore SRF05
BZ1: Buzzer senza elettronica
U1: PIC16F876A (MF794MB)
U2: PNA4602
U3: LM2940
U4: TSOP1738
Q1: Quarzo 20 MHz
SW1: Deviatore a slitta
SW2, SW5: Microswitch

SV1, SV6: SERVOS03T
T1: BC547

Varie:

- Plug alimentazione
- Zoccolo 14+14
- Strip maschio 2 poli (2 pz.)
- Strip maschio 3 poli (6 pz.)
- Strip femmina 5 poli
- Porta batteria 4xAA
- Circuito Stampato

il robot, poi ridate tensione al circuito di controllo mantenendo premuto il pulsante P1: udrete due "beep", indicanti che è attiva la procedura di setup dei servo. Tale procedura prevede la regolazione fine e manuale della posizione di un servo alla volta. Il microcontrollore prevede che per primo sia impostato SV1; con i pulsanti P3 e P4 potete regolarne la posizione. Fatto ciò, premete P2 per passare ad SV2 e, sempre con P3 e P4, se occorre regolarne la posizione; ripremete P2 per passare ad SV3 e così di seguito sino ad aver

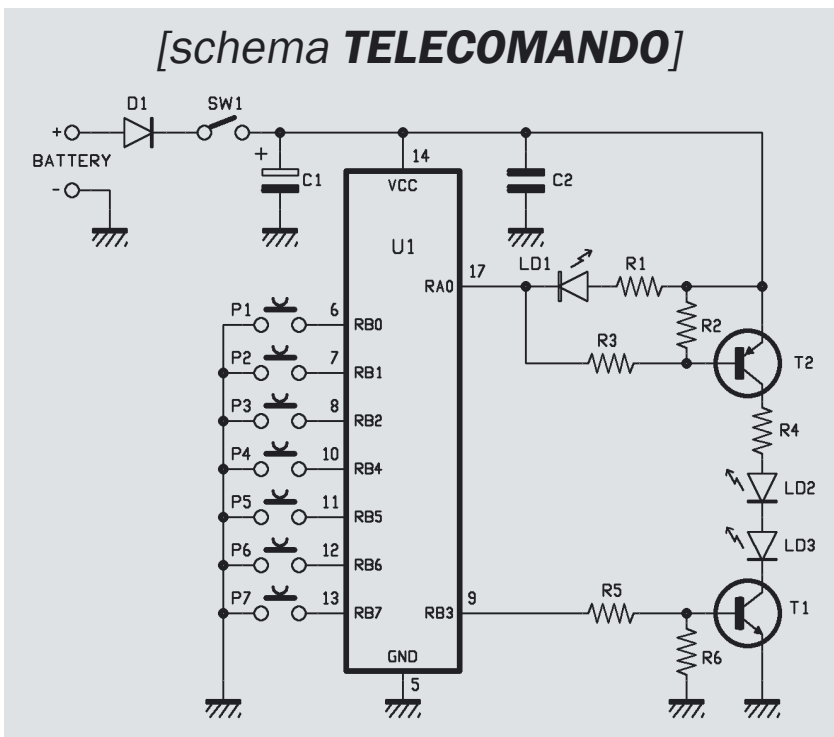
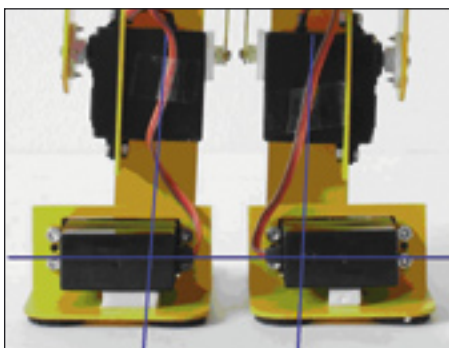
registrato tutti i servocomandi. Le impostazioni saranno salvate ogni volta, nel senso che ad ogni pressione di P2 il micro memorizzerà la posizione attuale per il servo cui si riferisce il passaggio appena concluso; questo vuol dire che quando ritenete di aver concluso la regolazione fine della posizione dei servocomandi fuori posizione, potete premere P2 e spegnere il robot in qualsiasi momento, certi che le impostazioni non verranno perse. Quindi se dovete ritoccare solo il secondo servo, accendete il robot con P1 premuto, attendete le due note

acustiche, quindi, se la posizione del primo servo va bene, premete P2 e passate all'impostazione del secondo servocomando; fatto ciò premete P2 e non preoccupatevi di procedere fino al sesto: togliete alimentazione alla scheda di controllo. Riaccendete il robot ed assicuratevi che sia tutto a posto; eventualmente, potete ripetere questa procedura ogni volta lo riteniate opportuno. La corretta posizione prevede una postura perfettamente diritta e verticale con solo il busto leggermente piegato in avanti, cosa da noi studiata per ottenere un migliore

bilanciamento dei pesi. Una volta acceso il robot, potete comandarlo manualmente tramite i pulsanti P1÷P4, i quali hanno il seguente significato:

- P1 = cammina in avanti;
- P2 = calcia con piede destro;
- P3 = ruota sul posto;
- P4 = stop e ritorno in posizione di riposo.

Ricordate che la posizione di riposo corrisponde al robot dritto in piedi, con tutte e due le gambe allineate. Notate che se il vostro automa ha la tendenza a cadere all'indietro, dovete provare ad inclinarne il busto maggiormente verso la parte anteriore; se tende ad essere troppo chinato in avanti, inclinatelo indietro. In entrambi i casi la postura può essere regolata agendo sul gioco dei fissaggi tra le varie parti delle gambe o, come già detto, mediante la procedura di regolazione fine della posizione dei servocomandi. A causa delle inevitabili dissimmetrie, durante la camminata il robot potrebbe non proseguire in linea retta, ma deviare leggermente a destra o a sinistra; in questo caso è necessario ritrarre la posizione di neutro dei servo. Se, ad esempio, dovesse tendere a sinistra, impostate i due servocomandi SV1 e SV2 per mantenere il robot leggermente inclinato sulla destra. Il robot cammina in maniera ottimale su di una superficie liscia, come ad esempio un parquet, un pavimento piastrel-



lato o una scrivania da ufficio; se deve muoversi su un piano ruvido o con asperità, i suoi movimenti appariranno leggermente impacciati. I feltrini applicati sotto i piedi consentono di avere un presa migliore, pur permettendo la giusta scorrevolezza durante i movimenti.

IL TELECOMANDO

Dover premere i pulsanti sulla scheda principale è assai scomodo, così come avere lunghi cavi per fornire comandi da PC; per questo abbiamo dotato il nostro robot di un telecomando a raggi infrarossi, che permette di impartirgli gli ordini a qualche metro di distanza senza dover andare vicino a disturbarlo.

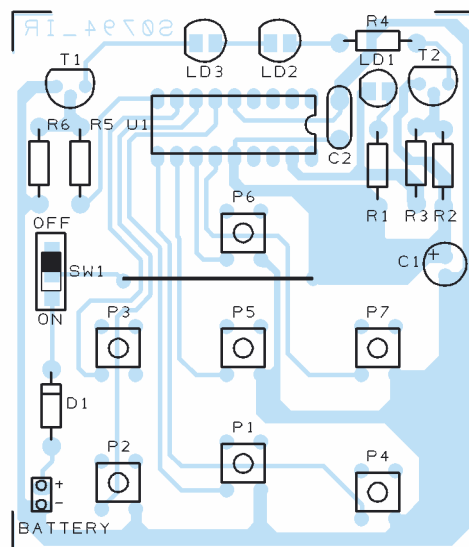
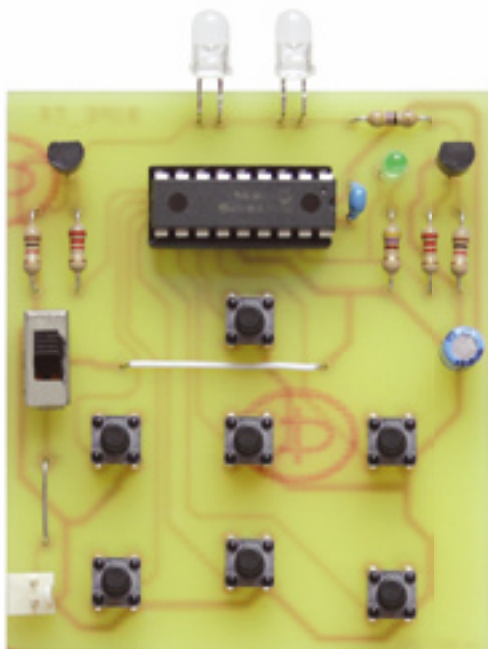
Abbiamo evitato moduli radio perché più costosi di un dispositivo IR e peraltro inutili, visto che nel nostro caso una portata

di alcuni metri è più che abbastanza. Sulla scheda di comando è sufficiente installare un modulo ricevente molto economico e facilmente reperibile, che riceve gli impulsi di un semplicissimo trasmettitore basato su di un micro; dallo schema elettrico corrispondente, potete notare la semplicità del circuito, che impiega un semplice PIC e pochi altri componenti.

Per comprendere come funziona il microcontrollore del trasmettitore dovete considerare che per impartire un comando bisogna tipicamente trasmettere dei dati mediante impulsi a infrarossi, impulsi che non possono essere semplici lampi in quanto, se così fosse, il ricevitore sarebbe troppo esposto a disturbi causati da altri telecomandi o da qualsiasi altra fonte IR. Quindi abbiamo previsto per la codifica, che consiste in una precisa modalità di invio dei raggi infrarossi, modalità che il ricevitore deve ovviamente conoscere. Nel nostro caso trasmettiamo degli impulsi utilizzando un segnale di frequenza fissa, modulato da impulsi più lenti che

Fig. 8 - Se il robot tende a camminare verso destra o sinistra, basta registrarne la posizione a riposo inclinandolo un po', rispettivamente, a sinistra o destra.

[piano di **MONTAGGIO** del **TELECOMANDO**]



Elenco Componenti:

R1: 470 ohm
R2: 1 kohm
R3: 2,2 kohm
R4: 47 ohm
R5: 2,2 kohm
R6: 1 kohm
C1: 10 μ F 16 VL elettrolitico

C2: 100 nF multistrato
T1: BC337
T2: BC327
U1: 16F818 (MF794IR)
SW1: Deviatore a slitta
LD1: led 3 mm verde
LD2, LD3: diodo IR L53-F3C

P1 ÷ P7: Pulsanti NA da c.s.

Varie:

- Zoccolo 14+14
- Strip maschio 2 poli
- Porta batteria 4xAA
- Circuito Stampato

costituiscono i dati contenenti i comandi da impartire al robot; la frequenza di questa portante è 38 kHz, valore al quale opera la gran parte dei telecomandi usati nel nostro Paese.

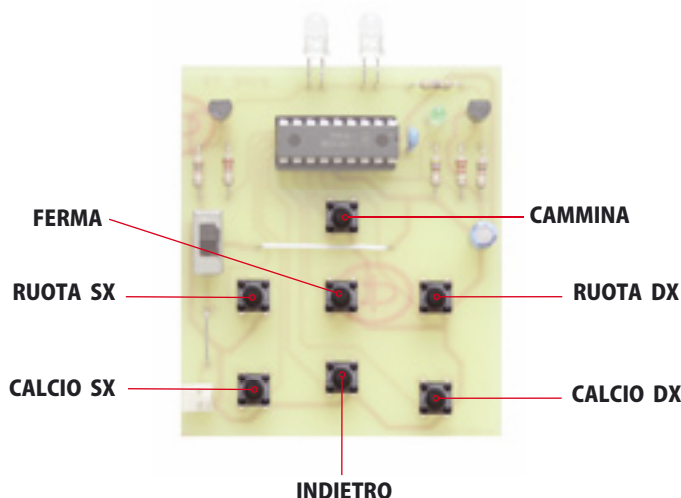
Il segnale che modula la portante è, in questo caso, un treno di impulsi a 2.400 baud, facilmente gestibile dal compilatore usato per programmare il PIC; quindi il sistema in sé risulta molto semplice da implementare. Nel nostro caso la portante viene generata dal modulo PWM interno al PIC, impostato per produrre un'onda con duty-cycle del 50 % e frequenza di 38 kHz, che viene resa disponibile al pin RB3. La modulante (ossia i dati a 2.400 baud) viene invece fatta uscire

dal pin RA0. Per generare il segnale infrarosso corrispondente alla tensione modulata è sufficiente comandare i due diodi D3 e D4, i quali, pur presentandosi come normali led, sono capaci di generare un fascio di luce all'infrarosso (non visibile all'occhio umano) come quello generato da un telecomando per TV. Per riuscire a vedere il led infrarosso che si illumina, potete usare una qualsiasi videocamera o telecamera allo stato solido (anche il telefono cellulare, se dispone di fotocamera o videocamera) e inquadrarlo mentre premete un pulsante: vedrete subito il lampeggio dovuto all'invio dei dati; ciò è possibile perché i sensori d'immagine CCD e CMOS,

oltre che al visibile sono sensibili anche all'infrarosso. Nel circuito del trasmettitore, il transistor Q1 è costantemente comandato dal segnale a 38 kHz, tuttavia i due led all'infrarosso sono percorsi da corrente solo se anche Q2 è in condizione, ovvero se RA0 è a livello logico basso. La particolare configurazione adottata permette di modulare in modo ON/OFF la portante, cosicché i diodi all'infrarosso ricevono treni di impulsi a 38 kHz ottenuti applicando e togliendo il segnale in base alla cadenza della componente digitale emessa dal PIC16F818 sul pin RA0.

In ricezione, dopo il modulo IR a tre piedini, si otterrà un segnale sul pin RC7 del PIC 16F876 iden-

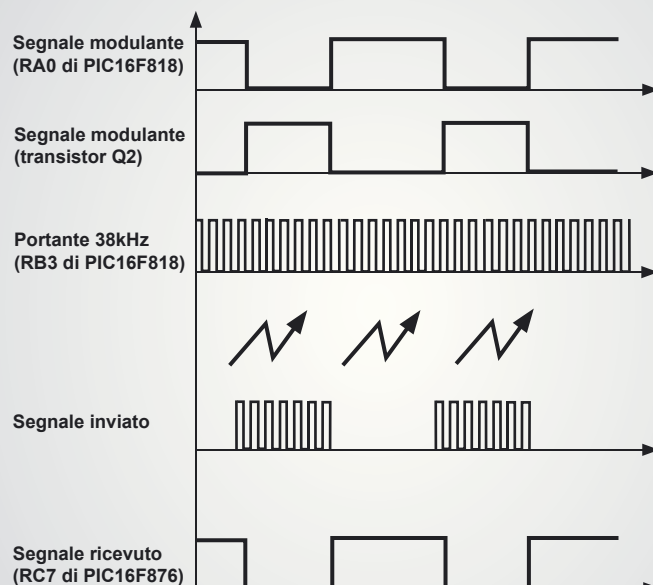
Disposizione dei pulsanti sul trasmettitore del telecomando.



tico a quello uscente dalla linea RA0 del PIC16F818 del trasmettitore. Nella trasmittente, il diodo D2 si accende in corrispondenza della conduzione del transistor Q2 e quindi monitorizza l'invio dei comandi mediante brevi lampi di luce. Riguardo all'alimentazione, anche per il trasmettitore del telecomando abbiamo previsto due possibili soluzioni: quattro batterie tipo stilo o mini-stilo, nel qual caso è obbligatorio saldare anche D1 per impedire che la tensione di alimentazione possa danneggiare il PIC; oppure tre batterie e in questo caso si può omettere D1, saldando al suo posto uno spezzone di filo di rame. Quest'ultima soluzione è quella adottata nel prototipo visibile nelle foto di queste pagine. A riposo, il trasmettitore assorbe una corrente irrisoria, pari a 650 μ A; questo vuol dire che potete lasciare collegate le pile senza alcun problema. Ma per i lunghi periodi di inattività, abbiamo comunque previsto un piccolo interruttore che scolga completamente la batteria dal circuito. Un così basso consumo a riposo è stato ottenuto bloccando la generazione della portante quando nessun pulsante è premuto e facendo in modo che il transistor Q2 sia normalmente interdetto.

IL FIRMWARE

Spieghiamo adesso, rapidamente, il firmware del trasmettitore del telecomando, che è più semplice di quello del microcontrollore che governa l'unità di controllo del robot, ma riteniamo sia molto utile a quanti siano interessati a realizzare un sistema di trasmissione a raggi infrarossi dal costo contenuto. Per abbreviare la spiegazione ci limitiamo a riportare gli elementi chiave del listato per PICBASIC necessari alla trasmissione IR; il listato completo è comunque a disposizione sul sito della rivista. Il tutto è facilmente adattabile a qualsiasi altro PIC diverso da quello da noi usato, a patto che integri il modulo PWM hardware (la gran parte dei PIC ha questa periferica). Le prime righe del listato definiscono il modo di funzionamento del generatore PWM hardware interno al PIC. Il programma principale esegue semplicemente il controllo di quale pulsante viene premuto; per semplicità è stato riportato solo P1, ma similmente avviene per gli altri pulsanti. A tasto premuto viene richiamata la funzione "Trasmette" che abilita la portante, invia il carat-



Correlazione tra il segnale dati prodotto dal PIC16F818 usato nel trasmettitore, quello modulante, la portante a 38 kHz, la componente pulsata trasmessa e i dati ottenuti nel ricevitore della scheda di controllo del robot.

tere corrispondente al pulsante premuto, quindi disabilita la portante.

È stato inserito un piccolo ritardo in modo che se un pulsante viene mantenuto premuto, il comando viene ripetuto all'incirca due volte al secondo. Se si preme più di un pulsante contemporaneamente, però, viene inviato il solo comando relativo al pulsante che risulta per primo nell'ordine previsto dalla procedura *main*. Per la trasmissione abbiamo usato la funzione SEROUT, che permette di inviare, utilizzando l'interfaccia standard RS232, il carattere corrispondente al tasto attivato; la modalità chiamata "0" imposta una trasmissione del tipo 2400,8,N,1 ovvero 2400 baud, 8 bit di dati, nessuna parità, un bit di stop. Dunque, i comandi inviati dal trasmettitore altro non sono se non caratteri ASCII espressi ciascuno con un byte binario che ne esprime i valori corrispondenti.

Notate che, per effetto di ciò, il micro nella scheda di controllo del robot riconosce i comandi sotto forma di caratteri, perciò è evidente che se si decide di con-

Listato TX IR

```

DEFINE HPWM1_TIMER 2      Hpwm ch1 usa timer 2
DEFINE CCP1_BIT 3         ' Hpwm pin bit
DEFINE CCP1_REG PORTB     ' Hpwm pin port

CH var byte                ' Carattere inviato

Main
  if P1=0 then
    ch="A"
    goto Trasmette
  endif
  goto Main

Trasmette
  hpwm 1, 127, 38000
  pause 10
  serout TX, 0, [CH]
  pause 10
  hpwm 1, 0, 0
  pause 300
  goto

```

trollare il "bipede" da computer, quest'ultimo dovrà inviare caratteri

ASCII; in teoria, basterebbe usare un emulatore di terminale abbinato alla porta seriale in uso per la connessione.

Detto ciò, proseguiamo con la funzione HPWM, la quale comanda direttamente il modulo PWM tramite tre semplici parametri: il primo specifica quale canale PWM viene usato (ciò è necessario perché il software può essere usato su molti PIC, alcuni dei quali dispongono di due o più moduli PWM...) mentre il secondo definisce il valore del duty-cycle, ovvero fissa il rapporto tra il tempo in cui il segnale rimane a livello logico alto e quello per il quale resta a livello logico basso. Notate, a riguardo, che un valore "0" indica che il segnale è sempre a zero, un "255" (risoluzione a 8 bit) indica che lo stesso è sempre a livello alto, mentre un "127" vuol dire un duty-cycle del 50 %, quindi il tempo in cui il segnale rimane a livello logico alto è uguale a quello in cui rimane a livello logico basso (onda quadra simmetrica).

Per chi volesse saperne di più sulla codifica del nostro telecomando a infrarossi, la Tabella 1 illustra l'abbinamento del

pulsante premuto al carattere inviato. Passiamo ora alla descrizione del firmware del micro della scheda del robot, dicendo innanzitutto

è spento, i servo non esercitano alcuna opposizione, tuttavia il tutto rimane in posizione, a meno di non spostarlo a mano. In condizione di *idle*, tutti i servo devono essere comandati con-

temporaneamente e costantemente, perché un servo da modellismo può mantenere una determinata posizione solamente se viene opportunamente comandato da determinati impulsi.

Ogni funzione è composta da una

fase di partenza e da una parte ripetitiva: ad esempio, per camminare ci sarà una prima parte relativa al movimento iniziale della prima gamba che si muove e poi un movimento ripetitivo relativo ad ogni passo, che viene richiamato più volte in sequenza. Il comando della posizione da raggiungere viene facilmente generato tramite la funzione PULSOUT presente nel compilatore PICBASIC, ma per ottenere movimenti fluidi e verosimili è necessario fornire tutte le coordinate intermedie relative al movimento desiderato dal punto iniziale a quello finale. Utilizzando un quarzo da 4 MHz, la risoluzione del comando PULSOUT

Tabella 1 - I caratteri corrispondenti ai comandi diretti al robot.

pulsante TX	Carattere inviato	Funzione
P1	A	Cammina indietro
P2	B	Calcina di sinistro
P3	C	Ruota sul posto a sinistra
P4	D	Calcina di destro
P5	E	Stop
P6	F	Cammina avanti
P7	G	Ruota sul posto a destra

che la maggiore difficoltà nello scriverlo è stata dover coordinare il controllo di sei servocomandi contemporaneamente, ognuno con velocità ed escursioni diverse. Dopo l'accensione della scheda di controllo, il firmware provvede a portare il robot in uno stato di *idle*, in cui è fermo in attesa di un comando. Premendo uno dei pulsanti sul robot o inviando dati dal telecomando o dal PC, vengono eseguiti specifici sottoprogrammi relativi alle funzioni richieste.

Noterete che anche quando il robot è in *idle*, tutti i servocomandi mantengono le posizioni e se cercate di muoverne uno, questo vi si opporrà. Se invece il robot

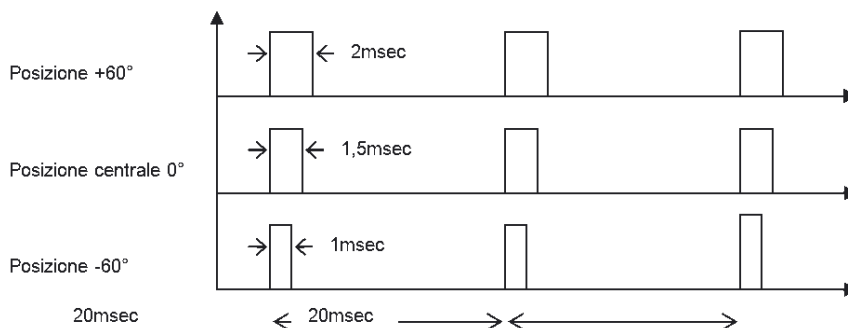


Fig. 9 - Correlazione tra la durata degli impulsi e la posizione assunta dal perno del tipo di servocomando utilizzato nel progetto.

è di 10 μ s, quindi per determinare la posizione di neutro useremo il valore 150, pari a 1,5 ms, che corrisponde alla durata degli impulsi richiesta dalla maggior parte dei servocomandi reperibili in commercio per posizionare al centro il proprio perno. Andando a modificare questo parametro possiamo far ruotare a destra o a sinistra il perno del servo: con il valore 100 la rotazione sarà di circa 60° e con 200 otterremo sempre 60°, ma in senso opposto. Questi impulsi devono ripetersi 50 volte al secondo, ovvero ogni 20 ms. La Figura 10 chiarisce meglio il concetto. La sintassi della funzione PULSOUT è la seguente:

PULSOUT *Pin, Period*

dove *pin* indica il piedino del PIC dal quale sarà fatto uscire l'impulso e *Period* il valore della durata dell'impulso, espresso in passi da 10 μ sec (se il quarzo è da 4 MHz). La Figura 11 mostra le tipiche posizioni di un servocomando standard in corrispondenza dei tre comandi corrispondenti alla massima rotazione a destra, a sinistra, nonché alla posizione centrale. È possibile variare la velocità con la quale vengono modificati i comandi e l'estensione del movimento; in questo modo è possibile far eseguire al servo movimenti molto fluidi e precisi nel punto esattamente voluto. Non spieghiamo tutto il firmware, peraltro molto lungo, ma passiamo alla descrizione del movimento attinente al "Calcio con la gamba destra"; anche in questo caso trascuriamo le impostazioni iniziali, che comunque sono visibili nel listato del file sorgente disponibile, assieme agli altri file del progetto, sul sito della rivista. La porzione di firmware corrispondente al

movimento in esame è descritta nel Listato 1. La posizione istantanea di ogni singolo servo è definita dalla variabile POSx (dove x è il numero del servo) mentre POSx_nt è la posizione di neutro di ogni singolo servo definita in fase di taratura. La posizione di neutro può essere impostata in un range che va da 1,4 a 1,6 ms (per l'impostazione grossolana occorre agire meccanicamente); ogni movimento avviene prendendo a riferimento proprio questo valore. Il programma è strutturato in modo che il perno di ogni servo venga fatto ruotare in una determinata posizione fornendo anche il comando relativo a tutte le posizioni intermedie. Con una risoluzione della larghezza di impulsi pari a 10 μ s, se, ad esempio, vogliamo portare un servo dalla posizione corrispondente a 1,48 ms verso quella relativa a 1,75 ms, dovremmo fornire, tramite la funzione *pulsout*, anche tutti i valori intermedi: 1,48 - 1,49 - 1,50...1,73 - 1,74 - 1,75 ecc., per un totale di 27 valori. Possiamo anche calcolare il tempo necessario allo spostamento, sapendo che ogni aggiornamento della posizione del servo avviene ogni 20 ms: nel nostro esempio, per 27 impulsi il tempo totale sarebbe poco più di mezzo secondo. Ad ogni ciclo FOR corrisponde un movimento di uno o più servocomandi di un certo angolo in un determinato numero di step. Maggiore è il numero di step, più lentamente sarà eseguito il movimento, maggiore sarà il valore finale della variabile POSx e tanto più ampia sarà l'escursione del movimento. Ad ogni nuovo calcolo della posizione istan-

tanea viene richiamata la funzione "Esegui", che consente di aggiornare il valore dell'impulso di comando appena calcolato. Il tutto è strutturato in modo che questa funzione venga eseguita all'incirca ogni 20 ms, secondo quanto previsto dalle specifiche dei servocomandi.

Se vogliamo che il robot dia

Listato 1 - La parte di firmware necessaria a far calciare il robot con il piede destro.

Listato 1

```

Calcias_DX
for i=1 to 30 ' oscilla a sx
  pos1=pos1_nt-i/2
  pos6=pos6_nt-i
  gosub esegui
next i

for i=1 to 25 ' prepara il piede dx
  pos4=pos4_nt-i
  pos5=pos5_nt+i
  pos6=pos1_nt-30+i
  gosub Esegui
next i

for i=1 to 25 ' calcia
  pos4=pos4_nt-25+45
  pos5=pos5_nt+25-45
  gosub Esegui
next i

for i=1 to 20 ' ritorna piede in posizione
  pos4=pos4_nt+20-i
  pos5=pos5_nt-20+i
  gosub Esegui
next i

for i=1 to 30 ' piedi paralleli
  pos1=pos1_nt-15+i/2
  pos6=pos6_nt-5+i/6
  gosub esegui
next i

goto main

Esegui
  pulsout SV1, pos1
  pulsout SV2, pos2
  pulsout SV3, pos3
  pulsout SV4, pos4
  pulsout SV5, pos5
  pulsout SV6, pos6
  pause 10
return

```



**Pulsout
SV1, 150**



**Pulsout
SV1, 100**



**Pulsout
SV1, 200**

Fig. 10 - Posizioni assunte dal perno del servocomando per vari parametri del comando Pulsout: in alto quella centrale, in centro la massima rotazione in un senso e in basso la rotazione possibile nel senso opposto.

un calcio con il piede destro, è necessario coordinare alcuni movimenti anche se non tutti contemporaneamente; analizziamo in dettaglio il firmware.

Oscilla a sinistra

Questa parte provvede a far inclinare il robot sulla sinistra, in modo da poter lasciare sollevato da terra il piede destro.

Per fare ciò è necessario comandare il piede sinistro (SV1) ed anche il piede destro (SV6); il movimento del piede destro con la sua inclinazione aiuta il robot a posizionarsi inclinato sul lato sinistro. Il servo SV6 dovrà avere escursione doppia rispetto il servo SV1. Il ciclo FOR calcolerà quindi 30 posizioni intermedie, portando il servo SV1 dalla posizione di neutro POS1_nt alla posizione POS1_nt-15, mentre il servo SV6 verrà portato nella posizione POS6_nt-30.

“prepara il piede destro

A questo punto occorre portare indietro il piede destro agendo sulla giunzione dell'anca e del ginocchio destri e, nel contempo, mettere il piede parallelo.

Il ciclo relativo FOR porta il servo SV6 dalla posizione POS6_nt-30 alla posizione POS6_nt-5, ovvero quasi parallelo al suolo, in quanto il robot è leggermente inclinato sulla sinistra.

Il servo SV4 viene portato in 25



passi nella posizione SV4_nt-25, mentre il servocomando SV5 viene posto nella posizione SV4_nt+25; il tutto si realizza simultaneamente.

“calcia”

A questo punto è necessario sferrare il calcio, il che si ottiene portando in avanti il piede destro il più velocemente possibile; per fare questo usiamo un ciclo FOR che posiziona immediatamente il servo SV4 a SV4_nt+20 e SV5 a SV5_nt-20. Il ciclo FOR in questo caso non viene usato per calcolare le posizioni intermedie, ma solo per lasciare il tempo necessario a sferrare il calcio; indicativamente possiamo dire che la nuova posizione viene comandata 25 volte e ogni volta dura circa 20 ms, per un totale di circa 0,5 s.

“ritorna piede in posizione”

Con questo ciclo FOR si fa in modo che la gamba ritorni in posizione di riposo. Il servo SV4 dalla posizione SV4_nt+20 viene portato nella posizione SV4_nt mentre il servo SV5_nt-20 viene riposto nella posizione SV5_nt.

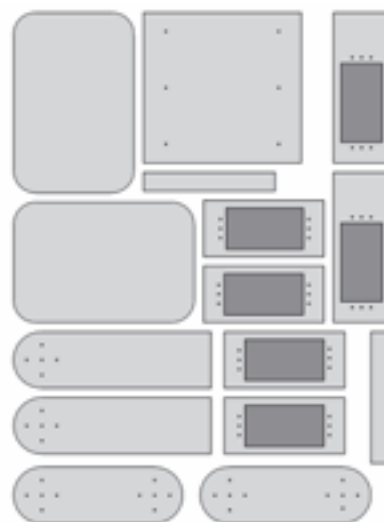
“piedi paralleli”

Il robot è ancora inclinato sul piede sinistro e bisogna riportarlo in posizione. I due servocomandi interessati si trovano nelle rispettive posizioni: SV1_nt-15 e SV6_nt-5. Il ciclo FOR in 30 step riporta i servo nella posizione di

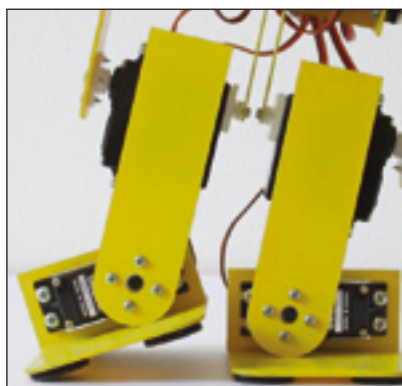
neutro SV1_nt e SV6_nt. I movimenti dei servocomandi per le altre funzioni li abbiamo determinati in maniera analoga; purtroppo, data la complessità dei movimenti, non esiste un sistema più semplice ed è necessario avere un minimo di conoscenze nel campo dell'informatica per comprendere completamente il software. La determinazione di ogni singolo movimento potrebbe essere spunto per qualche insegnante che voglia proporre un utile esercizio da sviluppare in qualche corso di informatica delle scuole superiori. Un metodo alternativo a quello qui descritto, al quale avevamo pensato, era costruire delle tabelle di dati contenenti le coordinate in sequenza di tutti i servocomandi, da richiamare quando occorreva compiere un certo movimento; per quanto più pronto, un metodo simile avrebbe richiesto un'occupazione di memoria troppo elevata, perché descrivere tutti i passi dei possibili movimenti richiede uno spazio non indifferente.

GESTIONE COMANDI ESTERNI

Quando il firmware è impegnato



Modello per la preparazione delle parti componenti il telaio del robot; l'immagine va ingrandita al 400 %.



Un robot davvero...spaziale



a determinare la nuova posizione dei servo e ad aggiornare gli impulsi di uscita corrispondenti, non c'è tempo di eseguire ulteriori funzioni, le quali causerebbero il rallentamento dei movimenti. Per verificare se è stato impartito un comando esterno da telecomando, abbiamo sfruttato il modulo USART interno al PIC, il quale, essendo un blocco hardware separato, non impegna direttamente la CPU.

Il modulo USART, opportunamente programmato, rimane in attesa (sul pin RC7/RX) dell'arrivo di un carattere seriale, avvisando tramite *interrupt* dell'avvenuta ricezione. Solo allora la CPU sarà disimpegnata dal programma principale e andrà a gestire l'arrivo del comando. Tra le funzioni svolte vi è anche quella di generare un picco "beep" di avvenuta ricezione di segnale dall'USART. I comandi possono essere impartiti anche tramite la porta seriale del PC, a patto che questa sia impostata per una trasmissione del tipo 2400,8,N,1; ad esempio con il semplice *HyperTerminal* di Windows oppure con qualsiasi altro software da voi

È di poche settimane fa (inizio novembre) la notizia che al Johnson Space Center di Houston (Texas, USA), la divisione Robot Systems Technology della NASA ha sviluppato Robonaut, il robot-astronauta, progettato per sostituire e/o affiancare gli astronauti umani nelle attività extraveicolari nello spazio, quali il montaggio, la manutenzione e la riparazione delle stazioni orbitanti. Il progetto, che ha anche risvolti di carattere militare, è condotto in collaborazione con DARPA (Defense Advanced Research Project Agency), la principale organizzazione di ricerca e sviluppo del Dipartimento della Difesa statunitense. Il robot, di cui sono stati realizzati due esemplari, è antropomorfo, ma privo degli arti inferiori, assolutamente inutili nelle operazioni EVA (Extravehicular Activity). La struttura umanoide del tronco e degli arti superiori è dovuta comunque al fatto che gran parte della tecnologia hardware dei veicoli spaziali è stata finora realizzata in funzione degli interventi diretti degli astronauti, e che pure gli strumenti e gli utensili per gli impieghi in ambiente EVA sono stati concepiti per la manualità umana. Le due braccia e le rispettive mani (a cinque dita) di Robonaut, ai cui movimenti si aggiungono quelli di torsione del tronco, sono azionate mediante 300 attuatori, coordinati da un sistema autonomo oltre che dai dispositivi per il telecontrollo remoto. Le capacità operative di Robonaut potranno superare in molti casi quelle degli astronauti, sia nelle missioni troppo lunghe e rischiose sia nei casi in cui siano richiesti forza, delicatezza, regolarità e precisione nell'uso degli strumenti. Gli ultimi test effettuati al Johnson Space Center di Houston, in una Stazione Spaziale simulata, sono stati tutti superati brillantemente. Al robot è stata aggiunta una gamba, con la quale ha passeggiato in un simulatore di stazione spaziale. Il comando avviene tramite una combinazione tra un'interfaccia di realtà virtuale e comandi verbali, che permette di gestire tutti i movimenti. Ma in un'era in cui i più facoltosi fanno a gara per andare nello spazio, pagando fior di quattrini ad agenzie nate appositamente, viene da chiedersi quanti astronauti già lo guardino con invidia, temendo di vedersi rubare il posto sulla prossima navicella in partenza per lo Spazio.



per il MATERIALE

Il firmware utilizzato in questo progetto può essere scaricato dal sito della rivista (www.elettronica.in.it). I due microcontrollori utilizzati sono anche disponibili già programmati al prezzo di 18,00 Euro (MF794MB) e 12,00 Euro (MF794IR). I servocomandi utilizzati (cod. 7300-SERVOS03T) costano 18,50 Euro cadauno mentre il sensore di distanza (SRF05) costa 26,00 Euro. I prezzi si intendono IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287
<http://www.futurashop.it>

creato. Sarà necessario preparare un cavo ad hoc che porti il segnale GND della seriale (pin 5) al pin 1 del connettore J8 ed il segnale TX della seriale (pin 3) al pin 2 di J8.

CONCLUSIONI

Ancora una volta abbiamo dimostrato le svariate sfaccettature dell'elettronica con un progetto veramente insolito ed accattivante. Anche in questo caso il progetto ben si presta ad essere applicato al mondo della didattica, perché coinvolge nozioni di meccanica, elettronica ed informatica. Chi ne avesse la possibilità potrebbe slegarsi

completamente dalla programmazione qui utilizzata ed affrontare in proprio, anche con altri mezzi informatici, l'interessante compito di far eseguire al robot qui proposto tutti i movimenti che ritiene più interessanti. Sebbene il firmware della scheda di controllo non sia dei più semplici da comprendere, la realizzazione e la messa in funzione di questo robot risultano assolutamente alla portata di tutti. Con un minimo di conoscenze informatiche è possibile comprendere al meglio il firmware e provare a cimentarsi nella creazione di nuovi movimenti e questo, ve lo assicuriamo, vi darà notevoli soddisfazioni. ■



Tra le tante idee regalo, abbiamo trovato e provato un prodotto in kit sicuramente interessante per i giovani sperimentatori: un giocattolo tecnologico, anche se chiamarlo così è probabilmente riduttivo...

BRACCIO ROBOT A 5 MOVIMENTI

..... a cura della REDAZIONE

In questo fascicolo a cavallo tra l'anno che sta finendo e quello che verrà, abbiamo deciso di dare ampio spazio all'argomento *robot*, presentando il progetto del nostro magnifico "bipede" e andando a cercare qua e là qualcosa che si trova già pronto in commercio; abbiamo così trovato un prodotto in kit di montaggio, quindi sicuramente molto interessante sul piano didattico e certamente adatto ai ragazzi che vogliano avvicinare quella complessa, affascinante ed avveniristica disciplina che è la robotica. Si tratta di un braccio robotizzato, capace

di compiere ben cinque movimenti e di sollevare un peso di 100 grammi, stringendolo tra le sue "dita" artificiali; ma non solo: sul vertice, ossia dove sono collocate le pinze (o dita, se preferite...) è situato un led a luce bianca che serve ad illuminare la zona che gli sta davanti e quindi aiuta a localizzare il punto dal quale afferrare gli oggetti. Insomma, il braccio può lavorare anche al buio, dato che il led utilizzato come proiettore ha un angolo di emissione abbastanza ristretto ed emette perciò luce concentrata, anche in virtù del fatto che è del tipo



Fig. 1 - Tutte le parti del kit. Gli elementi meccanici sono raggruppati e numerati.

ad alta efficienza. Il prodotto è realizzato dalla belga Velleman e commercializzato in Italia dalla ditta Futura Elettronica di Gallarate (VA) - www.futurashop.it. Per come è costruito, per i materiali impiegati e le sue capacità, può essere considerato un giocattolo tecnologico, ma definirlo giocattolo è un po' una limitazione; prescindendo dal peso e dalle finiture, che sono rapportate al target e al prezzo del prodotto, il braccio è costruito con tutti i canoni ed ha una struttura meccanica che rispecchia in tutto e per tutto quella dei bracci robot usati, ad esempio, nelle catene di montaggio delle industrie. Certo, qui si tratta di una versione miniatura e costruita con materiali idonei



Fig. 2 - Vista del gruppo motore con motoriduttore a quattro ingranaggi e due alberi ausiliari.

del modellismo, diciamo, animato. Poi, montare da sé il braccio robotizzato e vederlo in funzione, scoprire le sue notevoli possibilità, potrà essere motivo di divertimento, incentivo per intraprendere, più grandi, lo studio della robotica. Infatti, come già accennato, pur trattandosi di qualcosa di poco impegnativo, il braccio è in grado di compiere movimenti articolati e complessi, snodandosi per assumere praticamente qualsiasi posizione; le sue "dita" prensili gli consentono, inoltre, di afferrare un oggetto, una volta posizionatosi per raggiungerlo. L'insieme è simile a quello di un nostro arto superiore: la base è la spalla, il primo segmento corrisponde alla zona dell'omero e il suo snodo all'articolazione scapolo-omero; il secondo pezzo è l'avambraccio, che si muove su un motore che permette la piega del gomito. Infine, la testa è la mano e il suo punto di unione, comprendente il motore che la piega in alto e in basso è assimilabile al polso. La pinza simula le dita. I cinque movimenti previsti (rotazione sul piano orizzontale, avanti e indietro del braccio base, sollevamento e abbassamento del secondo pezzo di braccio, abbassamento e sollevamento delle

a sollevare o muovere oggetti di piccola massa. Ma lo scopo del kit non è certo montare un'automobile o saldare una rete metallica per edilizia; il kit ha una duplice utilità: innanzi tutto addestrare il giovane alla realizzazione di modelli meccanici, cosa che certo piacerà agli amanti



Fig. 3 - Base tonda di sostegno: gli anelli gialli contengono il motore che fa ruotare il braccio, sostenuto dal piatto nero

pinze, apertura/chiusura delle pinze) possono essere comandati simultaneamente, grazie a una scatola di comando dotata di altrettante levette; a realizzarli, provvedono cinque motori elettrici a bassa tensione (per cui il tutto funziona a pile) ciascuno dotato di motoriduttore che conferisce forza e fa muovere le parti con la velocità adatta, ma che, oltretutto, mantiene la posizione impedendo il ritorno quando al motore viene tolta corrente. Inoltre, ogni motoriduttore ha, sull'ingranaggio dell'alberino del motore, un sistema a frizione che evita il sovraccarico quando il motore stesso non regge lo sforzo richiesto dal movimento. Insomma, come abbiamo accennato, nella progettazione del braccio sono state adottate soluzioni che tipicamente si trovano in apparati robotizzati di tutt'altra fattezza. Per conoscere e farvi conoscere meglio il prodotto, ne abbiamo preso e montato uno, seguendo le istruzioni (peraltro estremamente dettagliate e chiarissime fornite nel manuale in dotazione al kit) di montaggio ed anche il nostro intuito, così da scoprire trucchi e problematiche pratiche. In queste pagine descriviamo i

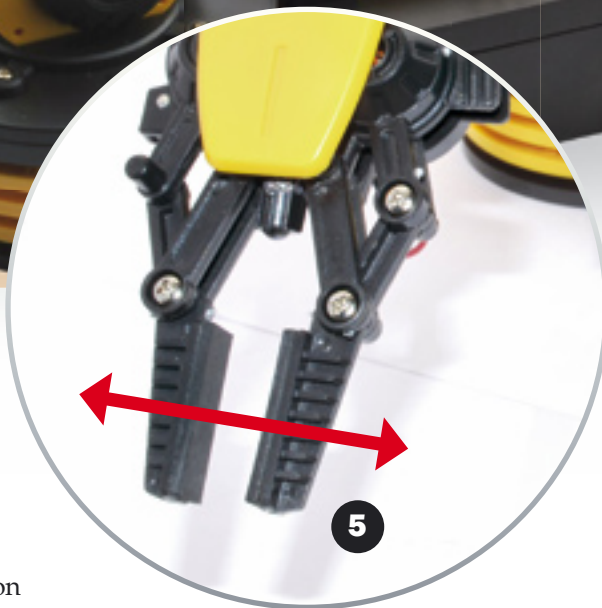
I movimenti del braccio

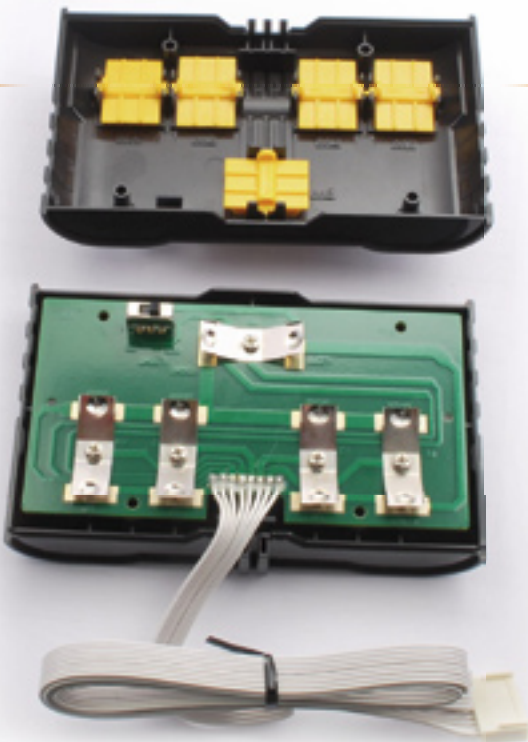
Grazie all'opportuna collocazione dei cinque motori di cui dispone e alla struttura meccanica che riflette, semplificata, quella del braccio dell'uomo, il braccio robot Velleman può compiere diversi movimenti, che lo rendono molto flessibile e adatto ad afferrare e posizionare oggetti entro un ampio campo d'azione.

punti salienti della realizzazione e le prove in campo. Si parte togliendo dalla confezione tutti i componenti, ben ripartiti in buste divise per tipologia (ingranaggi, motori ecc.) e facilmente identificabili (Fig. 1). Per mettere insieme il braccio robot bisogna innanzi tutto assemblare i cinque gruppi-motore con motoriduttore, che hanno ciascuno un involucro in plastica contenente un certo numero di ingranaggi (Fig. 2); ciascuno va chiuso con tre viti autofilettanti, che sono quelle previste dal manuale, dopo aver introdotto i due alberini ed aver infilato su questi i quattro ingranaggi richiesti. Infine si devono inserire i motori, facendo entrare la vite senza fine posta all'estremità dell'albero nell'ingranaggio vicino. Notate che le ruote dentate non sono tutte uguali, ma nessun problema: si distinguono in base al colore che hanno e il manuale di istruzioni dice quali usare. Montati i motori, occorre preparare la base, che contiene il vano delle pile di alimentazione (quattro elementi torcia) da un lato e il basamento di rotazione dall'estremità opposta; nella scatola base (vano pile) bisogna collocare i contatti a molla per le pile, quindi portare i fili sopra al coperchio, dove va fissata la scheda che realizza le connessioni con la scatola dei comandi e tutti i motori, oltre al led che sta sul vertice del braccio. Notate che le clip delle pile devono essere posizionate in modo che quella più grande stia dal lato opposto a quello della base rotante (base del braccio) mentre gli altri due, ciascuno con i fili, stanno dalla parte opposta; di questi due, quello con la molla va inserito in modo che si affacci alla parte piatta del contatto

1. rotazione base;
2. avanti e indietro braccio base;
3. su e giù parte superiore del braccio, (piega del gomito);
4. su e giù pinza;
5. apre e chiude la pinza.

grande, mentre quello piatto deve rivolgersi alla molla dello stesso contatto grande. Altrimenti diventa difficile bloccare le pile. Il supporto rotante è composto, oltre che dalla base, da alcuni anelli da impilare, all'interno dei quali trova posto il primo motore con





motoriduttore (Fig.3); va chiuso superiormente dopo aver applicato il coperchio alla scatola delle batterie. Per chiudere questo coperchio bisogna, prima, collegare al circuito stampato delle connessioni il cavo in arrivo dalla scatola dei comandi; quest'ultima va assemblata avvitando sullo stampato le mollette che realizzano i contatti (Fig. 4) e che, ad assemblaggio completato, verranno premute contro i contatti stagnati della basetta ramata dalle cinque leve

Fig. 4 - Scatola dei comandi aperta: alla base si trova il circuito stampato che contiene i contatti fissi; quelli mobili sono in una busta a parte e vanno avvitati in modo da sovrapporsi, ma senza toccare le piazzole.

di comando. Collegata la scatola comandi e chiuso il coperchio del vano batterie, si può chiudere anche il coperchio circolare della base rotante, sulla quale bisogna poi applicare l'ultimo ingranaggio del motoriduttore.

A parte, bisogna assemblare il braccio vero e proprio, iniziando dal pezzo più in basso, secondo le istruzioni; alla sua parte più bassa deve essere collocato il pezzo tondo che va sulla base rotante, incastrato nell'ingranaggio del motore di quest'ultima. Questo pezzo contiene una sua ruota dentata fissa, sulla quale deve essere calettato l'ingranaggio del secondo motore, che trova collocazione nell'apposito vano alla base del braccio; questo motore permetterà l'inclinazione in avanti o indietro del braccio robot. Ora è il momento di mettere insieme il secondo segmento del braccio, che si compone di

due motori
con motoriduttore,
uniti da

due liste esterne da avvitare per realizzare un insieme compatto (Fig. 5); a fine montaggio, l'ingranaggio di un motore entrerà nella ruota dentata della prima parte del braccio, in modo da permettere il sollevamento e l'abbassamento del secondo segmento (il motore fa ruotare l'ingranaggio su quello fisso del pezzo base). L'altro ingranaggio si incastrerà in quello della testa prensile, quella con le pinze e il led, in modo da permetterne il sollevamento e l'abbassamento e dare, così, un ulteriore movimento di flessione; questo snodo rende ancor più "elastico" il braccio robotizzato e consente l'esecuzione di movimenti molto precisi simili a quelli della mano. La testa prensile (Fig. 6) si assembla con l'ultimo motore, il cui motoriduttore va introdotto nel basamento (da fissare al vertice del secondo segmento del braccio e inserire nell'ingranaggio corrispondente) e collegato con gli appositi ingranaggi ai tiranti che azionano le dita o pinze; sempre sulla testa,



Fig. 5 - Assemblaggio della parte superiore del braccio: nella cava del motore va un ingranaggio che ingrani con quello della parte inferiore.

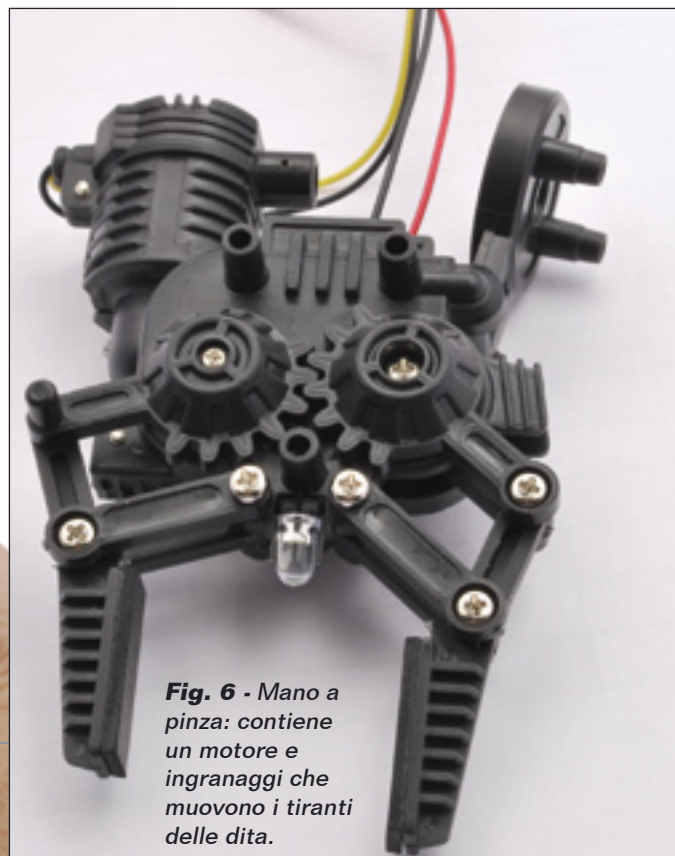


Fig. 6 - Mano a pinza: contiene un motore e ingranaggi che muovono i tiranti delle dita.



Fig. 7 - La coda della base contiene il vano batterie, chiuso da un coperchio che alloggia il circuito cui connettere i cavi in arrivo dai motori e dal led.

in un'apposita sede va collocato il led bianco, i cui fili, insieme a quelli del motore, devono scorrere sul secondo segmento del braccio e poi sul primo, fino a raggiungere il circuito stampato che realizza le connessioni, dove vanno inseriti negli appositi connettori (Fig. 7). Notate che ogni connettore femmina volante, corrispondente a un motore o al led, deve andare nella relativa coppia di punte, altrimenti le leve di comando non corrisponderanno alla nostra descrizione. Anche i fili dei due motori della seconda parte del braccio devono scorrere su questa e sul primo segmento, per poi arrivare al solito stampato. Per evitare che i collegamenti finiscano con l'intracciare i movimenti, bisogna fissare i fili mediante delle clip, peraltro fornite nel kit di montaggio. Ecco, assemblate tutte queste parti non resta che verificare se manca qualcosa e poi mettere subito alla prova il braccio, che a montaggio completato appare come in Fig. 8. Per azionare il robot bisogna agire sui rispettivi comandi, che sono dei semplici interruttori elettromeccanici che danno o tolgono l'alimentazione ai motori; si tratta di comandi senza posizione stabile azionati dalle levette, che, una volta rilasciate, tornano a riposo e aprono i circuiti. Osservate come non sia previsto alcun comando di accensione/spengimento generale, perché non ci sono circuiti che restano accessi a riposo, quindi anche la-

sciando le pile inserite non c'è rischio che si consumino; l'unico interruttore vero e proprio, a posizione stabile, è quello che accende e spegne il led sulla pinza. Per prima cosa proviamo a far ruotare il braccio sulla sua base: spingendo la leva corrispondente (quella centrale) a destra per ottenere la rotazione in senso orario e a sinistra per quella in senso antiorario. Ora abbassiamo l'intero braccio: per farlo, spingiamo indietro la terza leva da sinistra; se vogliamo alzare il braccio, spingiamo la stessa leva in avanti. Per abbassare o alzare la seconda porzione del braccio, ossia quella con i due motori, portiamo rispettivamente indietro o avanti la seconda leva da sinistra. Se desideriamo far piegare in basso o in alto la testa con le pinze, agiamo sulla leva di destra, spostandola, rispettivamente, indietro o in avanti. Non ci resta che provare il movimento della pinza: per chiudere le sue "dita" basta spingere in avanti la prima leva da sinistra, mentre per chiuderla occorre tirare indietro la stessa leva. Bene, imparati i movimenti base, perché non provare a fare qualcosa di più complesso? Ad esempio andare a prendere un oggetto a destra sul piano di appoggio del sistema robot? Allora, per prima cosa usiamo la leva centrale spostandola a destra fino a posizionare il braccio in modo che la pinza sia verticale al pezzo da prendere, poi abbassiamo l'intero braccio mediante la terza leva da sinistra. Per affinare il movimento, pieghiamo il gomito, cosa che



per il MATERIALE

Il braccio robot, contraddistinto dal codice KSR10, è disponibile in scatola di montaggio al costo di 62,00 Euro, IVA compresa. Il circuito comprende tutti i componenti meccanici ed elettromeccanici, comprese tutte le minuterie. Non sono comprese le batterie.



Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287
<http://www.futurashop.it>

otteniamo abbassando la seconda porzione del braccio (usiamo la seconda leva da sinistra) ed eventualmente anche la pinza, agendo sulla prima leva a sinistra. Poco prima di toccare l'oggetto, apriamo la pinza e stringiamola quando è in posizione per afferrarlo. Con una simile manualità, perché non divertirsi a prendere le palline decorative e metterle sull'albero di Natale? ■

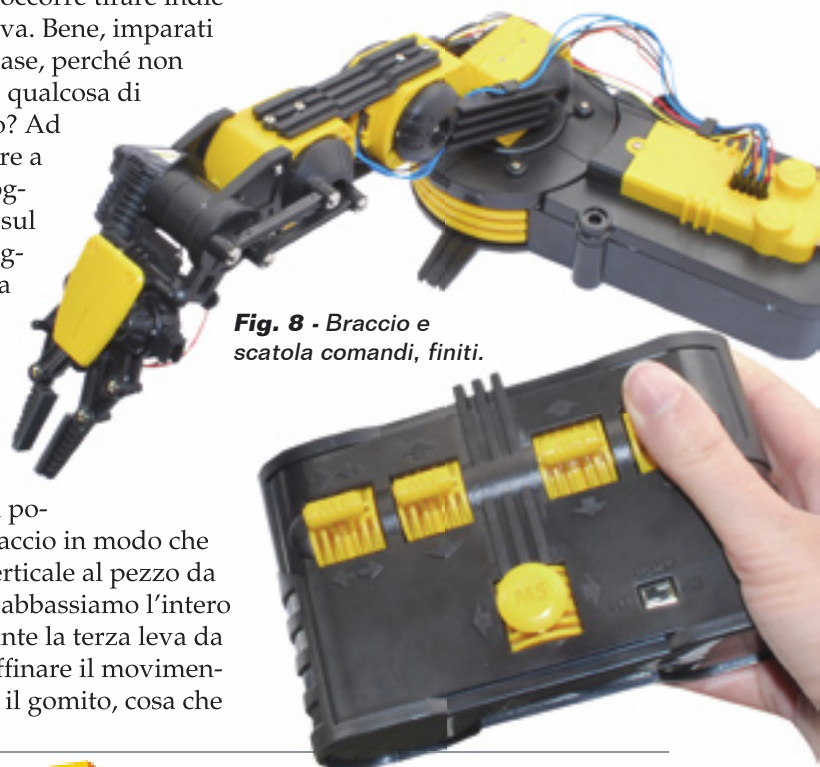


Fig. 8 - Braccio e scatola comandi, finiti.



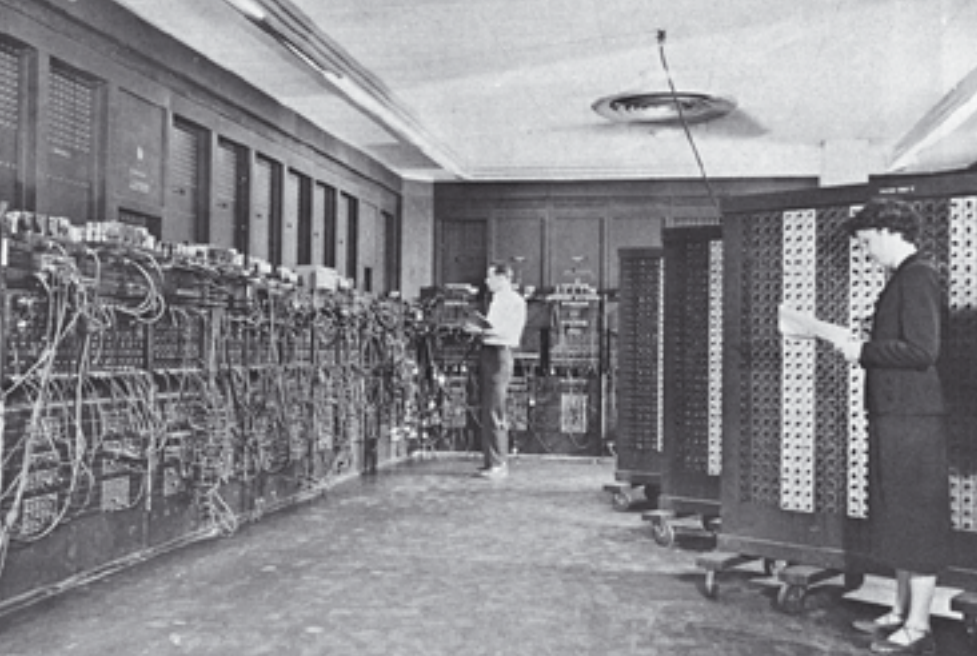
Dal PERSONAL COMPUTER al WORLD WIDE COMPUTER

Il Personal Computer come lo conosciamo oggi, da macchina sofisticata e quasi intelligente, sta per trasformarsi in un banale terminale di un immateriale Computer Globale a cui tutti saranno connessi tramite le reti a larga banda. Quali le conseguenze per la nostra vita quotidiana e per il modo di operare delle aziende?

..... di ARSENIO SPADONI

Fino ad un centinaio di anni fa la capacità di effettuare calcoli e memorizzare i relativi risultati era prerogativa del cervello umano tanto che con la parola "computers" venivano indicati gli impiegati che effettuavano calcoli e risolvevano equazioni, aiutati solamente da regoli meccanici. La prima vera macchina in grado di elaborare informazioni – il tabulatore a schede perforate – fece la sua comparsa nel 1890 in occasione del censimento della popolazione statunitense di quell'anno. In seguito queste macchine vennero adottate dalle

grandi aziende per elaborare, raccogliere e archiviare informazioni. Nel 1924 la più grande società del settore, sotto la guida di Thomas J. Watson, cambiò nome in International Business Machine (IBM) segnando, di fatto, l'inizio dell'era dell'Information Technology (IT). Da allora il compito di calcolare, classificare e archiviare passò dal cervello umano alle macchine, prima esclusivamente meccaniche, poi elettromeccaniche ed infine elettroniche. E' del 1937 quello che, sebbene ancora elettromeccanico, può essere considerato il primo calcolatore digitale

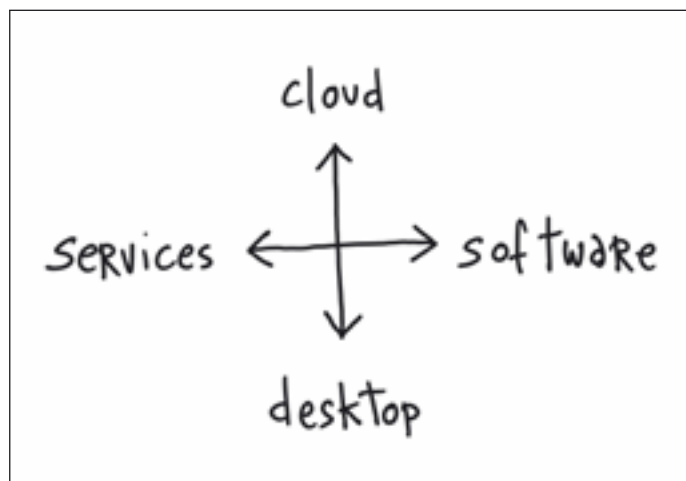


Ideato da John von Neumann, nel 1946 venne realizzato L'E.N.I.A.C., il primo grande computer che impiegava componenti attivi. La macchina pesava 30 tonnellate, occupava una superficie di 140 mq, utilizzava 19.000 valvole ed effettuava qualche migliaia di operazioni al secondo.

della storia: il Mark I, ideato da Howard Aiken, era infatti in grado di eseguire operazioni con numeri di undici cifre decimali. Una importante evoluzione rappresentò la sostituzione dei relè con le valvole; nel 1946 entrò in funzione l'E.N.I.A.C., concepito da John von Neumann, che veniva programmato con schede perforate ed effettuava una elaborazione batch ad una velocità di qualche migliaia di operazioni al secondo. Questa macchina utilizzava 19.000 valvole, pesava 30 tonnellate ed occupava una superficie di 140 metri quadri. Questi calcolatori erano un po' come i supercomputer di oggi: solo importanti aziende ed enti di ricerca potevano permetterseli. Vennero poi gli altrettanto costosi mainframe a cui erano collegati videoterminali costituiti da semplici tastiere e monitor e che utilizzavano, al posto delle valvole, i transistor e, in seguito, i circuiti integrati ed i microprocessori. Successivamente, con la miniaturizzazione, nacquero i minicomputer i quali, grazie alle loro dimensioni, potevano essere collocati su una scrivania. A poco a poco, queste macchine

si trasformarono nei PC che tutti noi conosciamo, macchine che, dotate di appositi programmi (software), sono in grado di effettuare, autonomamente, qualsiasi tipo di calcolo, elaborazione e memorizzazione. Nelle aziende i Personal Computer sono solitamente connessi ad un server centrale che ha lo scopo di memorizzare le informazioni comuni e di gestire la rete interna e, con l'avvento di Internet, anche l'accesso al "mondo esterno". In ogni caso il PC presente sulla scrivania di tutti gli impiegati (ed anche della maggior parte delle case di tutti noi) è in grado di effettuare, se dotato di adeguato software, qualsiasi applicazione. In altre parole la capacità di elaborazione delle informazioni viene soddisfatta dalla macchina stessa grazie al processore presente all'interno ed al software che ne coordi-

na il funzionamento. L'insieme delle macchine, degli accessori e, soprattutto, del software, costituisce il mondo dell'Information Technology (IT) che è la linfa vitale di qualsiasi azienda. Attraverso la rete informatica interna passano tutte le informazioni relative al magazzino, agli acquisti, alle vendite, alla contabilità ed alla fatturazione. E' del tutto evidente che questa configurazione è assolutamente inefficiente dal momento che la capacità computazionale complessiva dei PC installati in qualsiasi azienda viene sfruttata solamente in minima parte, al pari delle risorse software che sono ridondanti; su tutti i PC, ad esempio, è sicuramente installato il pacchetto Office: ma da chi e quante volte al giorno viene usato? In questa situazione le società di IT si sono sviluppate rapidamente allargando a dismisura il proprio giro di affari nonostante la riduzione dei prezzi dell'hardware. Con l'avvento di Internet, ma soprattutto con la disponibilità di reti di comunicazione a larga banda, le cose stanno iniziando a cambiare. Oggi i dati possono viaggiare alla velocità della luce ed è possibile - da remoto - erogare a ciascun terminale tutta la



potenza di calcolo necessaria. Ma non solo. E' anche possibile assicurare l'elaborazione delle informazioni nel rispetto di un programma prescelto nonché la memorizzazione e l'archiviazione dei dati. Se le informazioni possono fluire alla velocità della luce attraverso la rete, che differenza c'è se vengono elaborate localmente o da un potentissimo computer distante migliaia di chilometri? Oppure se vengono salvate sulla memoria di un computer remoto anziché sull'hard disk del server in fondo alla stanza? Quasi vent'anni fa l'attuale boss di Google Eric Schmidt (che allora lavorava in Sun) affermava che *"Quando la rete consente la stessa velocità del processore, il computer si svuota e si disperde attraverso la rete stessa"*.

Negli ultimi anni l'evoluzione tecnologica ha reso possibile la realizzazione di reti con simili prestazioni, che hanno raggiunto sia le aree metropolitane che i paesini più sperduti, grazie anche all'evoluzione delle soluzioni wireless radio. Ormai anche durante i nostri spostamenti possiamo restare collegati alla rete grazie alle soluzioni Wi-Fi, WiMAX e UMTS. La disponibilità di connessioni veloci ha reso possibile la grande diffusione di



L'interno e l'esterno (in basso) del più potente server farm di Google che si trova a The Dalles, una località dell'Oregon. Si stima che il motore di ricerca più famoso del mondo utilizzi complessivamente nei suoi centri oltre 2 milioni di server.

Internet così come l'abbiamo conosciuta finora; dopo questa prima fase, tuttavia, qualcosa di più importante sta prendendo corpo, qualcosa che cambierà radicalmente l'approccio al computing:

la potenza di calcolo migrerà dai nostri PC verso enormi data center, i programmi si trasformeranno in servizi che utilizzeremo solo quando ne avremo bisogno ed oltre al PC useremo svariati dispositivi portatili, tutti connessi alla rete.

Nel suo bellissimo ma inquietante saggio *"Il lato oscuro della Rete"*, Nicholas Carr paragona questa trasformazione a quella che ha visto la nascita delle grandi centrali elettriche. Alla fine del secolo XIX tutte le aziende producevano al proprio interno l'energia elettrica che serviva a fare muovere i macchinari mentre numerosissime piccole centrali elettriche di quartiere fornivano l'energia necessaria per l'illuminazione delle case e delle



strade. Nessuna azienda, a quel tempo, si sarebbe mai sognata di dismettere i propri generatori per rifornirsi di elettricità all'esterno. Cosa sarebbe successo - si domandavano gli industriali di allora - se il produttore di energia elettrica avesse interrotto la fornitura? La fabbrica avrebbe chiuso! Fu per merito di Edison, ma soprattutto del suo allievo Insull, che comprese per primo la grande inefficienza di questo sistema, se nel giro di pochi anni vennero costruiti enormi centrali elettriche in grado di fornire energia a buon mercato sia alle grandi industrie che ai privati cittadini. Grazie alle economie di scala, il prezzo dell'elettricità si abbassò drasticamente al punto che le aziende dovettero dismettere i propri generatori e collegarsi alle grandi centrali per mantenere il proprio livello di competitività. Il risparmio fu enorme: oltre a pagare l'elettricità meno della metà, le aziende non dovevano più investire ingenti capitali nella realizzazione di propri generatori elettrici. In sostanza pagavano soltanto quello che consumavano: l'elettricità era così diventata una *utility*. Come sempre accade

in presenza di trasformazioni radicali, l'elettrificazione delle città e l'elettricità a basso costo contribuirono alla nascita ed allo sviluppo di nuovi prodotti ed applicazioni e delle relative industrie e società di servizi determinando, ad esempio, la disponibilità di una vasta gamma di elettrodomestici ed un rapido sviluppo dei trasporti ferroviari e delle metropolitane. Per contro, interi comparti produttivi vennero spazzati via. Nel suo libro, Carr cita l'industria del ghiaccio, che prelevava questo prodotto nelle regioni artiche da dove, con grandi navi, veniva trasportato nei principali porti, e poi, via terra, distribuito un po' ovunque. Ebbene, con l'avvento dei frigoriferi elettrici, questa attività scomparve completamente. Oggigiorno -silenziosamente- sta succedendo la stessa cosa con il mondo dell'Information Technology; l'insieme dell'hardware, delle macchine e soprattutto del software, si sta trasformando in quello che qualcuno ha chiamato "nuvola" o insieme di nuvole: la potenza di calcolo sta diventando sempre più immateriale e viene



consumata dove e quando necessario. Nel cosiddetto *Cloud Computing* si stanno fondendo, come in un gigantesco computer universale, PC, telefoni cellulari, videogiochi, fotocamere digitali, telecamere, controlli remoti, ebook e sistemi di sicurezza. Tutto ciò comporterà un cambiamento radicale del modo di operare delle aziende, una profonda trasformazione del mondo dell'Information Technology e, per quanto riguarda tutti noi, un significativo cambiamento delle nostre abitudini. Ma anche nuovi rischi per la nostra privacy e la nostra sicurezza e conseguenze ancora tutte da valutare sulla società, sulla cultura e sui media. E' opinione condivisa che nel giro di pochi anni il *computing* diventerà una *utility* come l'energia elettrica, il gas o il telefono: in questo caso, tuttavia, il problema sta nella differente natura delle informazioni rispetto all'elettricità o al gas.

Esattamente come cent'anni fa con la nascita delle centrali elettriche, non possiamo considerare ciò a cui stiamo assistendo come una semplice evoluzione, per quanto significativa, del *computing*; si tratta piuttosto di una trasformazione epocale i cui effetti sulla società sono ancora tutti da comprendere. Ciò comporterà sicuramente la scomparsa di interi settori produttivi ed il forte ridimensionamento di altri. Se, sicuramente, il mondo dell'Information Technology è destinato a ridimensionarsi, altri



Per utilizzare i servizi messi a disposizione dal World Wide Computer utilizzeremo, oltre al tradizionale PC, tantissimi altri dispositivi, prevalentemente con connessione wireless. Oltre agli smart phone che già oggi consentono la navigazione in Internet e la lettura e l'invio di posta elettronica, utilizzeremo lettori digitali come Ebook Reader di Plastic Logic (a destra) e memorizzeremo i video e le foto in tempo reale sui nostri archivi remoti con fotocamere dotate di connettività wireless.

settori potrebbero seguire la stessa sorte o, addirittura, scomparire del tutto. Cosa succederebbe, ad esempio, all'industria della stampa se, come è molto probabile, gli Ebook Reader come quello rivoluzionario di Plastic Logic, si diffondessero rapidamente facendo scomparire giornali e riviste? E cosa accadrebbe all'industria della carta o, più semplicemente, alle edicole che vendono tali prodotti?

Se la "nuvola" - alla cui nascita stiamo assistendo - fa pensare a qualcosa di impalpabile che ci circonda, nel mondo reale questa complessa entità fa capo a tre elementi, distinti e molto tangibili:

- le infrastrutture;
- i servizi;
- i terminali.

A questi ultimi andranno a sommarsi un numero sempre crescente di sensori e dispositivi vari connessi alla rete e funzionanti in maniera completamente autonoma, ovvero senza la presenza dell'uomo.

L'INFRASTRUTTURA

Ogni settimana un container scarica 2.500 server presso il nuovo data center della Microsoft di

Northlake vicino Chicago. Si tratta di macchine poco più potenti di un normale PC che vengono rapidamente connesse alla rete elettrica, alle condutture d'acqua per il raffreddamento ed alla rete dati. Il software di gestione e controllo viene scaricato ed installato automaticamente e le nuove macchine si integrano con quelle già installate per incrementare la potenza di calcolo e la capacità di memorizzazione. Una volta ultimato, il nuovo data center della Microsoft utilizzerà ben 400.000 server. Sono "fabbriche" come queste che garantiscono e garantiranno sempre di più la capacità di calcolo necessarie ad aziende e privati sparsi in tutto

il mondo. Dal punto di vista costruttivo, l'aspetto più significativo di questi enormi centri di calcolo è il consumo di energia e la disponibilità di una fonte efficace di raffreddamento. Non a caso il più grande data center di Google è sorto a The Dalles, una località dell'Oregon che si trova in una zona ricca di corsi d'acqua in cui l'energia elettrica costa poco. Una località, tra l'altro, che si trova vicino ad un potente nodo Internet indispensabile per garantire una connessione con elevata ampiezza di banda (altro requisito fondamentale). Simili *data farm* sono in funzione sparsi un po' ovunque nel mondo; ad oggi si calcola che solamente Google utilizzi più di due milioni di server. Ma Google non si ferma qui; i suoi dirigenti stanno infatti ipotizzando la realizzazione di alcune navi con centinaia di migliaia di server a bordo, navi che trarranno l'energia dalle onde del mare ed utilizzeranno l'acqua per raffreddare i PC. Altre importanti società hanno realizzato o stanno per realizzare propri *data farm*, da Yahoo! ad Amazon: tutti con l'intento di rendere praticamente infinita la capacità computazionale della "nuvola". Allo stesso tempo sono già numerose le società e gli enti che stanno smantellando i propri centri di calcolo spostando verso



Un libro da non perdere

Avvincente come un romanzo, questo saggio racconta l'evoluzione dell'Information Technology e di Internet, dagli albori dell'era informatica fino all'attuale sviluppo degli enormi server farm che garantiranno nei prossimi anni servizi e capacità computazionale a tutti. Un viaggio che traccia un parallelo tra l'evoluzione dell'industria elettrica fino alla sua trasformazione in utility e quello dell'industria informatica, sul filo delle grandi trasformazioni economiche e sociali che tale sviluppo ha prodotto finora e soprattutto su quelle che il World Wide Computer determinerà in futuro. Una visione decisamente pessimistica su tutti i fronti, da quello economico a quello culturale, una serie di conseguenze negative che spesso - sostiene l'autore - la fede nella tecnologia ci porta ad ignorare o sottovalutare. Nella seconda parte del libro Carr affronta le ricadute del Cloud Computing sulla società: il rischio dell'impovertimento culturale, il pericolo del terrorismo online, dello spionaggio e di altre inquietanti applicazioni, la schedatura dei nostri dati e delle nostre vite, il collegamento uomo-computer tra fantascienza e scienza. Un libro che tutti dobbiamo leggere per capire come lo sviluppo di Internet condiziona il nostro futuro.

Nicholas Carr, ex direttore di Harvard Business Review, è un dei più noti esperti di tecnologia e i suoi articoli compaiono su New York Times, Financial Times, Wired e molte altre testate. Nel 2004 ha pubblicato "IT doesn't matter", un famoso (e molto discusso) articolo - sviluppato poi in un libro dal titolo *Does IT matter?* - sui limiti dell'approccio strategico basato unicamente sulla tecnologia. Nel luglio 2008 ha lanciato una nuova provocazione dalle pagine della rivista Atlantic Monthly, con l'articolo "Google ci rende stupidi?".



la "nuvola" attività che necessitano di grande capacità di calcolo, alta velocità e tanto spazio di memoria: il tutto allo scopo di ridurre i costi delle infrastrutture IT, eliminando nel contempo anche quelli fissi di investimento. Al momento si tratta ancora di dati non particolarmente "sensibili"; questa migrazione, tuttavia, sta rapidamente crescendo sia in termini di quantità che di qualità. Nella versione più estrema dell'*utility computing*, la "nuvola" rimpiazzerà completamente le

prestazioni dei PC e dei centri di calcolo: tutte le attività di processo e di memorizzazione verranno erogate dalla rete informatica riducendo il PC ad un semplice terminale: tastiera più monitor. Sui computer di questi enormi data farm possono funzionare contemporaneamente applicazioni differenti, specifiche per ciascun utente. Ciò è reso possibile dalla cosiddetta "virtualizzazione" dell'hardware, ovvero nella possibilità di utilizzare il software per simulare l'hardware.

Pur trattandosi di una tecnica già conosciuta ed applicata sui mainframe IBM a partire dagli anni '60, è solamente con l'avvento del *Cloud Computing* e di questi enormi centri che la virtualizzazione ha assunto un'importanza fondamentale. Questa tecnica consente di sfruttare al massimo la capacità computazionale di ciascuna macchina spostandola tra le applicazioni a seconda delle necessità. Anche per questo motivo i costi del *computing* si sono abbassati enormemente. Non è un caso, ad esempio, che le società piccole e grandi che ospitano i milioni di siti Internet utilizzino questa tecnica suddividendo una macchina tra centinaia di utenti, riuscendo così a fare pagare poche decine di euro un servizio tanto sofisticato.

DAL SOFTWARE AI SERVIZI

Se i cambiamenti nell'hardware sono così profondi, ancor di più - per quanto possibile - lo sono quelli che riguardano il software che, da applicazione che "gira" su uno specifico Personal Computer, si sta trasformando sempre di più in un servizio che viene erogato dalla rete, molte volte in forma del tutto gratuita. Sempre più spesso i provider di *computing* sono in grado di offrire agli utenti tool che consentono anche ai non esperti di creare e personalizzare proprie applicazioni. Questo nuovo approccio al software (noto come SOA, Service Oriented Architecture) può essere meglio compreso paragonando un cibo precotto acquistato al supermarket (vecchio modello) ad un ristorante in cui un cameriere prende le ordinazioni, un cuoco prepara il cibo ed un addetto alle pulizie si occupa di mantenere lindo il locale (nuovo modello). Il ristorante rappresenta l'applicazione

che si ottiene scegliendo quali e quanti oggetti (cameriere, cuoco e addetto alle pulizie) aggregare: l'utente non deve fare altro che scegliere tra oggetti differenti per avere l'applicazione personalizzata. Ma oltre a ciò, ed anche al fatto che ormai tutto il software viene scaricato tramite la rete, la più importante innovazione è rappresentata dalle applicazioni web-based, ovvero dalle applicazioni che "girano" e che sono state sviluppate per essere eseguite in rete. Ciò ha richiesto la creazione di alcuni standard che hanno rappresentato per l'*utility computing* quello che i caratteri mobili di Gutenberg hanno significato per la stampa. Nel XV secolo la stampa era una attività già nota da tempo, ma ogni volta che si voleva creare una nuova pagina, questa doveva essere incisa ex-novo; con i caratteri mobili di Gutenberg creare una nuova pagina risultava infinitamente più semplice. Se per un attimo ci soffermiamo a pensare cosa ha rappresentato nei secoli seguenti l'invenzione di Gutenberg per l'economia e la cultura, forse riusciremo ad intravedere quali trasformazioni comporteranno le applicazioni web-based. Ad esempio, oggi giorno per creare un sito o un blog esistono decine

di differenti applicativi on line che ci consentono di personalizzare le nostre pagine: da Drupal a DBlog, da CMS a WordPress (uno dei più diffusi): tutti assolutamente user friendly, e molto differenti dai vecchi strumenti come Dreamweaver che molti di noi hanno utilizzato sinora per creare i propri siti Internet. E per la gestione delle foto? Possiamo salvarle sui tantissimi siti di condivisione immagini come Flickr ed effettuare il fotoritocco tramite servizi web come Phixr, Picnic, XMG image, Pikifix e tanti altri. Così facendo le nostre foto elaborate e memorizzate su Flickr appariranno anche sul nostro blog. E se volessimo inserire anche dei video? Beh, in questo caso la soluzione la conoscono anche i bambini: memorizzeremo i video su YouTube e, con un semplice collegamento, li renderemo disponibili anche sul nostro blog o sul nostro sito. Anche i brani musicali e le nostre top list potranno essere incluse con altrettanta facilità. Bene. Ma come possiamo fare conoscere il nostro blog e mantenere i contatti con i visitatori? Niente di più facile: semplicemente dando loro la possibilità di essere informati delle novità implementando un servizio RSS. E se per caso

volessimo avere qualche introito da questa attività? Basterà aprire un account associato al servizio AdSense di Google: ogni volta che un visitatore cliccherà su uno di questi link, Google ci riconoscerà una piccola quota della pubblicità pagata dal suo inserzionista. Tutto ciò facendo ricorso a programmi "che girano" in rete, ovvero con quelli che ormai possono essere considerati a tutti gli effetti dei servizi. E' probabile che nei prossimi anni

RM ELETTRONICA
Forniture per hobbisti, scuole e industria

Distributore autorizzato:

FUTURA ELETTRONICA **ELSE Kit** **NUOVA ELETTRONICA**

Il tuo punto di riferimento per l'elettronica a ROMA

Via Val Sillaro 38 • 00141 Roma • Tel: 06-8104753


COMUNE di SCANDIANO

CENTRO FIERISTICO



30^a MOSTRA REGIONALE ELETTRONICA SCANDIANO (RE)

14/15 FEBBRAIO 2009

Orari:
Sabato 9-18,30
Domenica 9-18

Biglietti:
Intero euro 7 - Ridotto euro 4
 Ragazzi fino a 12 anni ingresso gratuito

Info: 0522/764302
www.fierascandiano.it
entefiere@comune.scandiano.re.it

telefonia
 componentistica
 computer
 hi-fi car
 radiantismo cb/om
 videoregistrazione
 mercatino delle pulci
 radioamatoriali



anche la maggior parte delle aziende elimineranno le proprie infrastrutture IT ed i propri software utilizzando applicazioni web-based (che potranno essere configurati facilmente in funzione delle proprie esigenze) e sfruttando la capacità computazionale della "nuvola". Ciò, oltre tutto, renderà le applicazioni accessibili ovunque e con strumenti anche molto differenti dal Personal Computer. Potremo accedere ai nostri archivi, ai nostri dati, al nostro album fotografico sia dal PC dell'ufficio che da quello di casa, ma anche dal cellulare e dall'ebook: ovunque e in qualsiasi momento.

SEMPRE PIU' INTERATTIVI: INTERNET DELLE "COSE"

A mano a mano che questo insieme di servizi e di capacità di calcolo diventerà sempre più diffuso e capillare, ad essa faranno capo sia una vasta gamma di sensori e attuatori senza alcuna intermediazione umana, sia una serie di dispositivi (prevalentemente wireless) che ci consentiranno di interagire con la *nuvola* in tutti i modi possibili. Oltre agli smart phone che già oggi consentono la navigazione in Internet e la lettura e l'invio di posta elettronica, utilizzeremo lettori digitali come Ebook Reader di Plastic Logic, memorizzeremo i video e le foto in tempo reale sui nostri archivi remoti con fotocamere dotate di connettività wireless, le nostre autovetture saranno munite di navigatori in grado di aggiornare il percorso in funzione del traffico e delle condizioni delle strade e terremo sotto controllo i nostri parametri fisici. Moltissimi oggetti saranno dotati di dispositivi in grado di connettersi alla *nuvola* dando luogo a quella che qualcuno ha definito come l'Internet "delle cose". Oggetti e persone faranno parte di un sistema globale di elaborazione delle informazioni le cui ricadute avranno ripercussioni molto difficili da prevedere. Chi ci ha provato - come Nicholas Carr - ne ha sottolineato gli enormi rischi, non solo per la nostra privacy, ma anche, più in generale, per l'economia, con la scomparsa di interi settori produttivi, la concentrazione della ricchezza e del potere nelle mani di gruppi sempre più ristretti di persone e una spinta all'impoverimento culturale. Sicuramente il *Word Wide Computer* mette a disposizione di ciascuno di noi nuovi strumenti per esprimerci e realizzarci ma allo stesso tempo consente a pochi di avere a disposizione strumenti di condizionamento e controllo molto sofisticati e molto più potenti di quelli attuali. L'effettivo impatto sulla società dipenderà da come si evolverà il confronto tra le nuove possibilità di libertà e la maggior capacità di condizionamento.

LOCALIZZATORE GPS/GSM MINIATURA

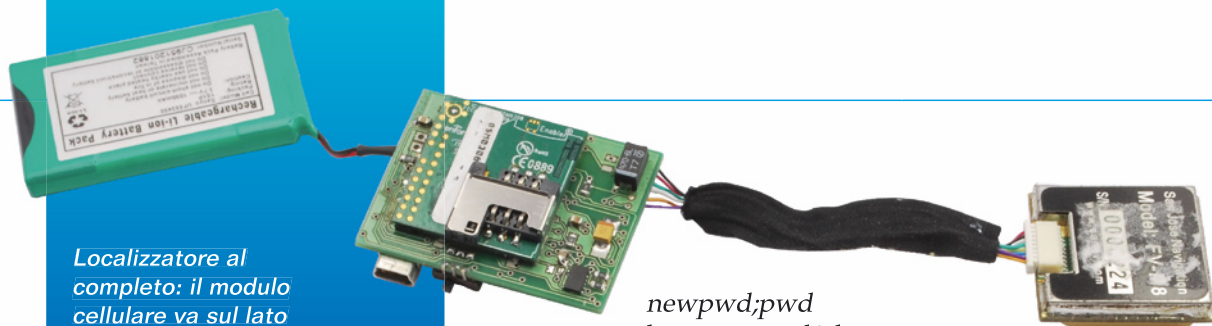


Memorizza e trasmette via SMS, e-mail o URL, i dati sulla propria posizione; alimentato a batteria, può economizzare al massimo l'energia spegnendo, nel caso, il ricevitore GPS. Ultima puntata.

..... di BORIS LANDONI

Nel fascicolo precedente abbiamo introdotto il progetto di un localizzatore GPS con cellulare GSM/GPRS, altamente miniaturizzato, in grado di rilevare e trasmettere a un computer, anche in tempo reale e con una connessione punto-punto, i dati sulla propria posizione o sul percorso compiuto. In particolare, il dispositivo rileva la propria posizione e la memorizza, secondo diverse modalità impostabili a piacere, periodicamente oppure in seguito a movimento o arresto; invia inoltre le coordinate periodicamente in automatico (*autoreport*)

su richiesta (*polling*) o a seguito di un evento che può essere il movimento e il conseguente arresto, ma anche l'entrata o l'uscita da una zona predefinita (*geofence*). Viste le caratteristiche tecniche, analizzati lo schema elettrico ed il funzionamento, è ora giunto il momento di vedere qualche nota costruttiva e scoprire in che modo si utilizza, conoscere le segnalazioni e l'uso dei pulsanti, apprendere quel che c'è da sapere sui comandi da remoto e sulla sintassi. In questa seconda ed ultima puntata vi parleremo anche del software di gestione sviluppato



Localizzatore al completo: il modulo cellulare va sul lato opposto a quello dei connettori. La batteria è al litio, da 3,7 V.

per l'ambiente Microsoft Windows, un programma in Visual Basic che permette di configurare il localizzatore e scaricare da esso i dati della memoria quando è collegato fisicamente mediante porta USB, ovvero di visualizzare la posizione in tempo reale, nella modalità real-time, quando, invece, il dispositivo è connesso punto a punto tramite modem GSM. Iniziamo subito parlando del set dei comandi da impartire, da remoto, mediante SMS, che avete già visto riassunti nella tabella pubblicata a pagina 74 del fascicolo n° 132 di Elettronica In.

I COMANDI DA REMOTO

Cominciamo con quelli che riguardano la gestione della lista dei numeri di telefono abilitati a effettuare la richiesta dei dati di posizionamento, numeri che sono 8 in tutto e che corrispondono altresì a quelli cui il localizzatore invia i predetti dati via SMS quando la funzione corrispondente è attivata. Prima di procedere, è il caso di fare una precisazione: alcuni comandi possono essere eseguiti solamente se corredati di password; altri, basta che giungano da telefoni i cui numeri siano presenti nella lista degli otto abilitati al comando e alla richiesta delle coordinate. Si noti che la password predefinita dal firmware (e comunque impostata automaticamente dopo un reset totale del sistema) corrisponde a 12345; è possibile sostituirla con una a piacimento (purché di cinque cifre) impartendo il comando del caso con un SMS contenente il testo PWD

newpwd;pwd
dove *newpwd* è la

password da memorizzare e *pwd* la password attuale.

Per memorizzare un numero, bisogna inviare un SMS contenente il testo *numxxxxxxxxxxxx;pwd*, dove *x* indica la posizione (1÷8) in cui salvare il numero rappresentato da *xxxxxxxxxx* e *pwd* è l'attuale password, necessaria se il comando SMS proviene da un telefono non in lista (serve comunque quando si memorizza il primo numero...) o se, pur arrivando da un telefono di quelli già memorizzati, la posizione specificata è al momento già occupata da un altro numero. Ogni numero può essere lungo al massimo 19 caratteri e deve comprendere il prefisso di nazionalità (per i cellulari italiani è +39).

Per cancellare il numero in una posizione della lista, il comando da usare è *NUMx;pwd*, dove *x* è la posizione (1÷8) e *pwd* l'attuale password; è anche possibile interrogare il dispositivo per conoscere tutti i numeri memorizzati: basta impartire il comando da SMS *NUM?;pwd*.

Una volta salvati i numeri, bisogna definire quali sono quelli cui il localizzatore deve inviare gli SMS contenenti le coordinate della posizione in modalità a seguito di interrogazione o di movimento o fermata: il comando corrispondente è *SMS xxxxxxxx;ON*, dove *xxxxxxx* sono le posizioni dove stanno i numeri. Ad esempio, *SMS1357;ON* imposta che a ricevere gli SMS siano i numeri nelle posizioni 1, 3, 5, 7. Ogni comando impartito cancella la precedente configurazione per la posizione cui è riferito.

Analogo è il comando che definisce a quali indirizzi di posta elettronica, fra quelli memorizza-

ti, il dispositivo debba mandare le e-mail contenenti i link o le coordinate: *EML xxxxxxxx;ON*. Ad esempio, *EML1456;ON* impone che quando viene ricevuta una chiamata il localizzatore invii le e-mail agli indirizzi che stanno nelle posizioni di memoria 1, 4, 5 e 6. Per impostazione predefinita, alla prima accensione e dopo il reset, il localizzatore invia e-mail ed SMS a tutti i numeri e indirizzi memorizzati al momento.

Passiamo adesso ai comandi che riguardano l'attività correlata con il rilevamento del movimento del localizzatore: il primo è *TMN:xx* e definisce la sensibilità, ossia quale debba essere l'intensità dello spostamento che determina l'attivazione della funzione *movimento* e causa l'invio di SMS, URL o e-mail del caso; al posto di *xx* si deve definire la sensibilità, tra 00 (massima) e 80 (minima). *TMF:ss* è il comando che definisce, invece, l'intervallo di tempo trascorso il quale, se non viene più rilevato movimento, il microcontrollore ritiene fermo il localizzatore e attiva l'invio di SMS, URL ed e-mail; al posto di *ss* bisogna scrivere il tempo voluto, tra 00 e 59 secondi.

Impostati l'intervallo temporale e la sensibilità, affinché il dispositivo invii SMS, URL o e-mail bisogna attivare la modalità corrispondente: il comando usato per attivare è *TMA:ON*, mentre se la modalità è attiva e si vuole disabilitarla, bisogna usare il comando *TMA:OFF*. Notate che il comando TMA è ripetuto localmente dal pulsante P2: premendo brevemente quest'ultimo, si attiva la modalità corrispondente (cosa che viene segnalata da un lampeggio del led LD2 per un

secondo); una pressione prolungata (almeno 2 secondi) provoca la disattivazione (fa quel che fa *TMA:OFF*) condizione segnalata da un rapido lampeggio del solito LD2.

In ogni momento, con uno dei messaggi *TMF?*, *TMN?* o *TMA?*, è possibile interrogare il localizzatore sull'attuale impostazione della funzione di invio a seguito di movimento o fermata; in risposta si otterrà un SMS contenente i tempi, la sensibilità e lo stato (funzione abilitata/disabilitata). Veniamo adesso ai comandi che riguardano la registrazione in memoria dei dati di posizionamento: *REGM:ON* e *REGM:OFF* servono rispettivamente per attivare e disattivare la registrazione a seguito di movimento, rilevato secondo le modalità (accelerazione, tempo di attività/inattività) definite dai comandi appena descritti. *REGC:ON* e *REGC:OFF* attivano e disattivano, rispettivamente, la registrazione periodica, che è indipendente dal movimento. *REGC:mm/ss* imposta ogni quanti minuti e secondi (*mm* ed *ss* possono valere ciascuno da 00 a 59, rispettivamente minuti e secondi) debba essere memorizzata la posizione rilevata dal ricevitore GPS nel caso sia abilitata la registrazione periodica; in tale situazione, è possibile fare in modo che il tempo definito cambi quando l'accelerometro rileva il movimento del localizzatore. A tale scopo si usa il comando *REGM:mm/ss*, dove *mm* ed *ss* sono rispettivamente minuti (00÷59) e secondi (00÷59) trascorsi i quali viene effettuata una registrazione. Questo comando serve, ad esempio, per rendere più frequente la registrazione quando il localizzatore si muove. Infatti se, ad esempio, in condizioni normali si può accettare un punto memorizzato ogni ora (59

minuti e 59 secondi) quando il localizzatore prende a muoversi può essere necessario registrare un punto ogni 10 secondi.

Se non si ricorda l'attuale impostazione dei tempi e dei modi di registrazione, in ogni momento è possibile richiederla al



Anche in Italia il GPS vigila sul bestiame

I numerosi furti di bestiame nella zona collinare del territorio della provincia di Forlì-Cesena, hanno spinto la Prefettura a correre ai ripari: accanto alle iniziative volte a rendere più mirato il controllo del territorio da parte delle Forze di Pubblica Sicurezza (Carabinieri e Corpo Forestale dello Stato), in sede di Comitato Provinciale per l'Ordine e la Sicurezza Pubblica l'Amministrazione Provinciale di Forlì-Cesena si è impegnata nel cofinanziamento (con l'Associazione Provinciale degli Allevatori) di una sperimentazione di sistemi di controllo elettronico degli animali. Nello specifico, il progetto riguarda l'applicazione di collari contenenti un localizzatore GPS/GSM. Già in occasione delle Giornate Zootecniche della Moia (25/27 Luglio ad Alfero-Verghereto) si è tenuta sabato 26 luglio 2008 una dimostrazione pratica di utilizzo di collare dotato di GPS in grado di segnalare al telefonino o al computer dell'allevatore la posizione dell'animale e quindi l'eventuale allontanamento dall'area di pascolo. Il sistema è oltretutto dotato di un antisabotaggio, in grado di rilevare e segnalare a distanza possibili tentativi di manomissione del collare. Tale dimostrazione è avvenuta con il supporto del Laboratorio di Bioingegneria dell'Università degli Studi di Milano, che collabora alla realizzazione del progetto.

Quasi un anno prima un analogo esperimento, condotto da Coldiretti e facoltà di Veterinaria dell'Università degli studi di Milano, aveva visto impiegare in Italia collari con localizzatore presso la fondazione Ferrazzi e Cova di Villa Cortese.

Tab. 1 - Segnalazioni fornite dai LED montati nel localizzatore.

LED	Colore	Segnalazione
LD1	rosso	Fix ricevitore GPS: pulsa per 1 secondo ogni 2 di pausa se manca il fix; lampeggia rapidamente se il ricevitore ha agganciato un numero sufficiente di satelliti.
LD2	giallo	Resta acceso se il cellulare non si registra nella rete; in tal caso, dopo 15 secondi il micro ripete la procedura di avvio. Segnala lo stato della funzione di allarme a seguito di rilevamento del movimento: si accende per un secondo all'attivazione e fa un lampeggio al momento della disattivazione. Quando tale funzione è attiva, lampeggia insieme a LD3 se il sensore rileva movimento.
LD3	verde	All'accensione, resta acceso fisso fintantoché attende la chiamata per l'easy setup. Nel normale esercizio, lampeggia una volta ogni 30 secondi se il localizzatore è in sleep; più lentamente se il ricevitore GPS è acceso o il localizzatore sta inviando segnalazioni d'allarme. Lampeggia più frequentemente se il GPS viene spento dal micro. Quando è attiva la funzione di rilevamento del movimento, lampeggia insieme a LD2 se il sensore rileva movimento. Segnala, inoltre, lo stato della funzione geofence: si accende per un secondo quando la si attiva e lampeggia brevemente quando, invece, la si disabilita.
LD4	rosso	TX dati su USB
LD5	verde	RX dati su USB
LD6	verde	Indicatore di carica della batteria: si spegne quando la carica è completata.

localizzatore inviandogli l'SMS contenente il testo *REG?*; in risposta si ottiene un SMS con le impostazioni. Come accennato in precedenza, il ricevitore GPS viene acceso e spento dal microcontrollore all'occorrenza; è comunque previsto un comando da SMS, impartendo il quale si può forzare manualmente l'accensione o lo spegnimento del GPS. Si tratta di un SMS contenente il testo *GPS:ON* per accendere o *GPS:OFF* per spegnere. Vediamo adesso i comandi che definiscono gli identificativi del localizzatore: un SMS contenente il testo *IMEI?* dà in risposta a chi l'ha mandato il codice IMEI del telefono. Il comando *NAME:xxxx* permette di assegnare un nome all'apparecchio, cosa utile quando si devono gestire più localizzatori; *NAME?* interroga, invece, l'apparecchio per ottenere il suo nome identificativo. Tutti questi comandi possono essere inviati da numeri presenti in lista, ovvero da qualsiasi altro, purché corredati di password. Bene, visti questi comandi passiamo a quelli che definiscono che tipo di comunicazione debba effettuare il localizzatore a seguito di un evento o in autoreport: *AMI:x* definisce la modalità di comunicazione a seguito del

rilevamento di movimento o fermata e in esso al posto della *x* va scritto *S* se si desidera l'SMS, *E* quando si vuole l'e-mail, *U* per l'invio dell'URL, *3* se si desidera avere, invece, sia SMS che e-mail, *4* per inviare sia SMS che URL, *5* per l'invio di e-mail ed URL e, infine, *9* per ottenere l'invio sia di SMS, che di e-mail ed URL. *AAI:x* è il comando SMS per definire le azioni in autoreport (polling) e al posto della *x* va inserito uno dei parametri anzidetti per il comando *AMI:x*. Gli stessi parametri si applicano al comando *ARI:x*, che definisce le modalità di comunicazione nel caso il localizzatore venga interrogato con uno squillo, e all'*AGE:x*, con il quale si impostano le azioni associate alla funzione *Geofence* (uscita dalla zona o entrata nella stessa). Definito quale mezzo usare per inviare le coordinate del localizzatore a seguito di eventi o in automatico, possiamo stabilire, per ciascuno, il formato da utilizzare, in modo da ottenere ogni volta l'espressione più consona. *FORS:*

Per quanto riguarda P3, notate che la funzione corrispondente si attiva solo se, tramite i comandi da SMS o da PC, è stata già definita una zona.

x stabilisce la rappresentazione degli SMS e *FORE:x* quella delle e-mail. Per l'URL la rappresentazione è fissa. In ogni momento si può interrogare il sistema con il comando *SMS FOR?*, per conoscere quale formato è attualmente impostato per un certo mezzo. Per i comandi di impostazione, il parametro *x* vale 1 se si desidera avere le coordinate con il formato seguente (usato, ad esempio, dal localizzatore CT24) **imei=011107000061937&rmc=\$GPRMC,091844.000,A,4538.6398,N,00848.8521,E,0.83,0.00,230908,,,A*61,POLLING**, 2 se si vuole una rappresentazione di lettura immediata quale, ad esempio, questa:

Position Report:
DAY: 23/09
TIME: 09:32:46
SAT: 3
LAT:45 38 67.94/N
LON:008 48 75.77/E
SPEED: 0 Km/h
DIRECTION: 240 g
ALTITUDE: 210
GPS fixed
POLLING

Se il parametro vale 3, si ha una rappresentazione simile alla precedente, però con le coordinate espresse in formato decimale, come nell'esempio seguente:

Position Report:
DAY: 23/09
TIME: 09:22:30
SAT: 3
LAT: +45.6438516
LON: +008.8143466
SPEED: 0 Km/h
DIRECTION: 240 g

Tab. 2 - Funzioni dei tre pulsanti.

Tasto	Funzione
P1	Trasmette allarme (SOS) immediato a distanza.
P2	Attiva/disattiva allarme a seguito di movimento.
P3	Attiva/disattiva geofence.

ALTITUDE: 213

GPS fixed

POLLING

Con l'opzione 4 si ottiene l'indirizzo Web della pagina *Google Maps* corrispondente alla posizione del localizzatore; il messaggio è del tipo:

<http://maps.google.it/maps?f=q&hl=it&q=+45.6438300,+008.8145983>

GPS fixed

POLLING

L'opzione 5, invece, fa in modo che venga inviata una serie di dati nel formato:

Position Report:

FT710

2008/09/23 09:37:11

+45.6446333,+008.8126416

GPS fixed

L'invio contiene il nome del localizzatore, data ed ora attuali, oltre alle coordinate di latitudine e longitudine in formato decimale. Infine, scegliendo l'opzione 6 otteniamo messaggi contenenti dati nel formato tipico dell'*Handytracker*, ossia una stringa utilizzabile dagli smartphone con installato il software che permette di visualizzare direttamente sullo schermo la posizione del punto ricevuto; un esempio di dato inviati è:

6 (tipo TR102 esteso) formato necessario per poter lavorare con Handytracker:

?011107000061937,5,3,230908,093816,E00848.7571,N4538.6816,99,0,48,3,3,80!

Qualunque sia il formato scelto, i dati comunicati terminano sempre con la modalità di funzionamento che ha prodotto l'invio delle informazioni: *POLLING*

Il localizzatore può essere utilizzato con servizi di tracking online come il GPSTRACER di cui abbiamo parlato in questo stesso fascicolo, ovvero in abbinamento ad Handytracker (http://www.globalsat.com.tw/eng/product_detail_00000111.htm) per vedere i dati di localizzazione su PDA e palmari predisposti.

Per poter funzionare con GPSTRACER, il localizzatore può essere impostato in due modi: il primo prevede l'uso di SMS selezionando come numero di destinazione +393480009219 e come formato per l'invio il 5 (comando FOR5). Il secondo usa le e-mail e richiede che come indirizzo di destinazione sia definito GPS1@GPSTRACER.NET e che il formato impostato sia l'1 (comando SMS FORE:1). Dal sito GPStracer deve essere selezionato il localizzatore FT710-SMS o FT710-Email.

Quanto all'Handytracker, sul nostro sito si trovano tutte le informazioni del software per Smart Phone e PDA phone. Il localizzatore non necessita di una particolare configurazione: riconosce in automatico la richiesta da parte dell'Handytracker e risponde direttamente nel formato corretto.

L'FT710 riconosce i comandi per l'autoreport (pag. 28 del manuale scaricabile in pdf dal sito Web www.futurashop.it), per il report immediato e per il report ciclico (pag. 33 del solito manuale).

quando l'invio è richiesto da una telefonata o dall'apposito SMS, *AUTO* in autoreport, *GEOFENCE IN* e *GEOFENCE OUT* per la funzione *Geofence*, rispettivamente nel modo a rilevamento dell'ingresso in una zona o dell'uscita da essa, *MOVE* se l'invio deriva da movimento e *STOP* da arresto. Infine, *SOS* indica che il pacchetto di dati è stato inviato perché qualcuno ha premuto il pulsante (P1) dell'SOS. Inoltre, con le opzioni dalla 2 alla 5, il messaggio contiene l'indicazione dello stato attuale del fix del ricevitore GPS (GPS fixed se il fix è effettuato). Osservate che se si desidera l'invio dell'URL non sono previste opzioni, perché il formato è uno solo: quello descritto dall'opzione 1 (tipo il localizzatore commerciale CT24). Per quanto riguarda la modalità di autoreport, abbiamo previsto un apposito set di comandi: *AUTOM: ON* e *AUTOM: OFF* servono, rispettivamente, ad attivare e disattivare l'autoreport (invio periodico) solo se il localizzatore viene mosso; *AUTOC: ON* e *AUTOC: OFF* permettono, invece, di attivare e disattivare l'autoreport incondizionato (invio periodico autonomo). *AUTOM: hh/mm* definisce l'intervallo tra un invio e l'altro quando l'au-

toreport è subordinato al movimento del localizzatore, mentre *AUTOC: hh/mm* stabilisce l'intervallo tra due invii nell'autoreport indipendente. Infine, *AUTO?* è il comando che permette di verificare a distanza lo stato delle impostazioni per l'autoreport. In caso di notifiche a seguito di movimento, fermata, *SOS*, richiesta tramite SMS o chiamata, il localizzatore risponde con un messaggio contenente le coordinate salvate in memoria (ultimo punto FIX) inserendo NO FIX se si tratta di dati non aggiornati o FIX se, invece, i satelliti sono agganciati.

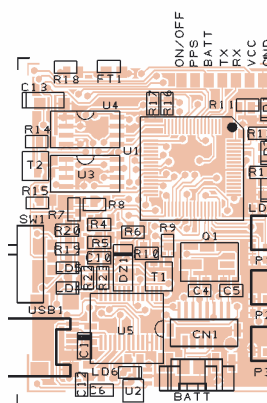
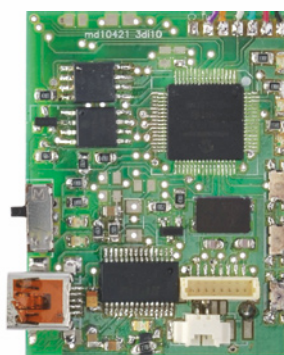
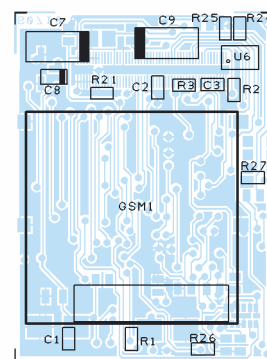
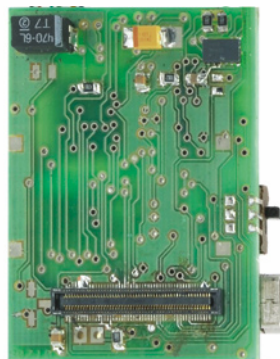
Adesso diamo uno sguardo al set di comandi riguardante le impostazioni del servizio e-mail: *EMAILx: indirizzo@indirizzo.yy* imposta gli indirizzi di posta elettronica cui il localizzatore invierà le e-mail; si possono definire al massimo otto destinatari e nella sintassi, *x* indica il numero del destinatario da impostare ed *indirizzo@indirizzo.yy* sta per l'indirizzo e-mail. Ad esempio, *EMAIL1: elettronica@elettronica.it* è il comando per salvare nella posizione 1 l'indirizzo di posta *elettronica@elettronica.it*. Per cancellare un indirizzo dalla lista, basta inviare il comando da SMS *EMAILx*, dove

R1: 330 ohm (0805)
 R2: 4,7 kohm (0805)
 R3, R10: 10 kohm (0805)
 R4÷R6: non montato
 R7: ÷R9: 4,7 kohm (0805)
 R11÷R13: 330 ohm (0805)
 R14: 4,7 kohm (0805)
 R15: 10 kohm (0805)
 R16÷R18: non montato
 R19, R20: 330 ohm (0805)
 R21, R22: 4,7 kohm (0805)
 R23, R26: 10 kohm (0805)
 R24, R25: 4,7 kohm (0805)
 R27: 0 ohm (0805)
 C1÷C3: 100 nF multistrato (0805)
 C4, C5: 15 pF ceramico (0805)
 C6: 100 nF multistrato (0805)
 C7: 470 µF 6 VL tantalio
 C8: non montato
 C9: 100 µF 6 VL tantalio
 C10: 100 nF multistrato (0805)
 C11: 4,7 µF 10 VL tantalio
 C12: 100 nF multistrato (0805)
 C13: 10 µF 10 VL tantalio
 U1: PIC18LF6722 (MF710)
 U2: MCP73831

U3, U4: 24LC1025-I/SM
 U5: FT232RL
 U6: MMA7455L
 LD1, LD4: led rosso (0805)
 LD2: led giallo (0805)
 LD3, LD5, LD6: led verde (0805)
 GPS: Sanav FV-M8
 DZ1: non montato
 Q1: quarzo 20 MHz (12SMX)
 T1, T2: BC817
 SW1: Deviatore a slitta
 FT1: non montato
 GSM1: GSM0308
 P1÷P3: Microswitch 90° SMD

Varie:

- Batteria al litio 3,7V-1000 mA
- Connett. CS100 pin per GSM0308
- Connettore Mini-USB
- Connettore Molex 2 poli 90° passo 1,25 mm
- Connettore 8 poli 90° passo 1 mm
- Circuito stampato



x è il numero della posizione che lo contiene. **EMAILFROM: PIPPO@PIPP0.IT** è il comando che definisce l'account di posta usato per l'invio delle e-mail; in pratica corrisponde all'indirizzo che il destinatario delle e-mail vedrà come mittente. Per cancellare il mittente memorizzato, basta il comando **EMAILFROM**, perché scrive "nulla" nelle posizioni di memoria corrispondenti. Il comando per verificare gli indirizzi memorizzati è **EMAIL?** e produce, in risposta, un SMS diretto al numero da cui è pervenuto, contenente gli indirizzi e-mail attualmente inseriti nel localizzatore. Il localizzatore risponde solo se il numero che ha impartito il comando è uno di quelli in lista. Ci sono poi i comandi per le impostazioni dell'accesso a Internet: l'APN si definisce con **GPRSAPN xxxxxxxxxx**, dove al posto delle x va inserito il parametro corrispondente al gestore del servizio Internet cellulare; per

Vodafone è **web.omnitel.it**. All'occorrenza, **GPRSAPN** cancella l'impostazione esistente. **GPRSUSR xxxxxxxx** serve a impostare il nome utente (username, da scrivere al posto delle x) e **GPRSUSR**, invece, cancella l'impostazione attuale. **GPRSPWD xxxxxxxxxx** definisce la password di accesso ad Internet e **GPRSPWD**, invece, cancella la password esistente. Infine, chi desidera verificare le impostazioni del servizio Web GPRS può usare il comando **GPRS?**, che in risposta produce un SMS diretto al numero che l'ha richiesto, contenente i dati del caso. Resta inteso che il localizzatore risponde solamente al comando proviene da un numero di quelli in lista o da uno qualsiasi a patto che l'SMS inviato contenga la password attualmente valida. Questi sono i comandi riguardanti l'accesso al Web; ora vediamo quelli per l'account di posta elettronica, effettivi anch'essi solo se provenienti da un

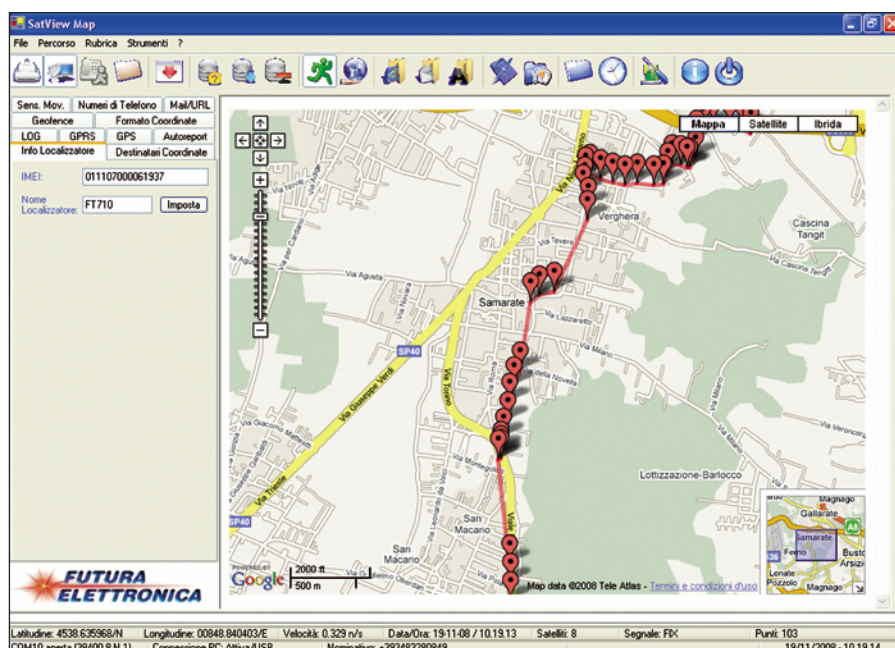
cellulare il cui numero sia uno degli otto della lista: **SMTPPRO: smtp.gestore.yyy** definisce il server SMTP (posta in uscita) necessario all'invio delle e-mail, mentre **SMTPPRO** cancella l'impostazione attuale. Infine, **SMTP?** interroga il localizzatore al fine di conoscere le attuali impostazioni per il server di posta in uscita; in risposta produce un SMS contenente tutti i dati richiesti. Passiamo ora a un'altra categoria di comandi: quella riguardante la definizione dell'indirizzo Web cui puntare quando l'invio dei dati di localizzazione è diretto a un URL: **URLDEST:nomeurl** imposta l'indirizzo della pagina, che può essere, ad esempio, **www.gpstracer.net/codifica.asp>80**, nel caso dell'URL da noi usato per le prove con il nostro servizio di localizzazione **www.gpstracer.net**. Il comando **URLDEST** provoca la rimozione dell'URL attualmente impostato e **URLDEST?** interroga il localizzatore al fine

di fargli comunicare l'indirizzo attualmente impostato; funziona esclusivamente se a impartirlo è un telefono il cui numero sia in memoria. Per quel che riguarda il *Geofence*, esiste un subset di comandi con cui definire i dati delle tre zone configurabili; il primo imposta la zona e i parametri che la contraddistinguono ed è *GEO: x/I-O/NOMEAREA/+LAT1/+LON1/+LAT2/+LON2*. La sintassi prevede che al posto di *x* vi sia il numero della zona in oggetto, *NOMEAREA* sia il nome con cui l'area verrà identificata nei messaggi inviati via SMS o e-mail, *LAT1/LON1* la coppia di coordinate di un vertice del quadrilatero descrivente l'area e *LAT2/LON2* la coppia di coordinate che corrisponde all'altro vertice del quadrilatero dell'area. Inoltre, *I-O* specificano, rispettivamente, se la funzione deve attivarsi all'entrata (*I*) o all'uscita (*O*) della zona definita. Ad esempio, *GEO:1/I/AREAUNO/+LAT1/+LON1/+LAT2/+LON2* definisce la prima zona, chiamata *AREAUNO*, imponendo l'attivazione quando il localizzatore entra in essa. Per attivare o disattivare la funzione *Geofence* si usano i comandi, rispettivamente, *GEO:ON* e *GEO:OFF*; invece, con *GEO?* si interroga il localizzatore per sapere quali sono le zone definite e se la funzione è attiva o meno. Anche i comandi di questa sezione sono effettivi se impartiti da un telefono il cui numero sia in lista. Per concludere la carrellata sui comandi riguardanti l'impostazione delle funzioni di invio dei dati, vediamo un subset che riguarda una funzione molto particolare: la

richiesta di trasmissione dei dati a un numero diverso da quello che impartisce il comando. Inviando un SMS contenente il testo *?0,IMEI,0,NumeroDestinatario!* il localizzatore manda al numero di cellulare indicato in *NumeroDestinatario* un messaggio contenente le coordinate; il comando deve essere completato con l'*IMEI* (da scrivere al posto del parametro *IMEI*) del modulo GSM/GPRS del localizzatore. Il comando *?1,IMEI,SECONDI,VOLTE,1,NumeroDestinatario!* serve invece a richiedere l'invio delle coordinate, a un telefono diverso da quello che ha fatto la richiesta, periodicamente per un certo numero di volte; il telefono cui arriveranno i dati si scrive al posto di *NumeroDestinatario*, mentre *SECONDI* e *VOLTE* sono, rispettivamente, l'intervallo e il numero di invii. *IMEI* è il solito codice *IMEI* del modulo GSM/GPRS del localizzatore. In ogni momento, con il comando *COO* è possibile richiedere le coordinate dell'attuale posizione; esso produce un SMS diretto al numero del telefono che l'ha

impartito. Il reset completo, che ripristina le impostazioni predefinite e cancella la lista dei numeri telefonici, si ottiene con il comando *RES;pwd*, dove *pwd* è la password attuale. Il sistema accetta messaggi multipli, ossia SMS contenenti più comandi ciascuno purché separati da una virgola; volendo, per risparmiare denaro è possibile disattivare le risposte ad alcuni comandi inserendo, nell'SMS, il comando *RISP*. Quest'ultimo fa in modo che, per i comandi che prevedono la conferma tramite SMS, il localizzatore non faccia partire i messaggi del caso. Notate che la configurazione del localizzatore qui descritta via SMS, può essere condotta anche da computer, mediante l'apposito software che descriveremo nei prossimi paragrafi. Notate altresì che il dispositivo prevede l'*Easy Setup*, ossia una procedura di memorizzazione automatica del primo numero telefonico della lista: dopo l'accensione e per circa tre minuti attende una telefonata, al sopraggiungere della quale salva in memoria il numero del

Appena avviato il programma Sat View Map, si presenta la finestra principale visibile qui accanto.



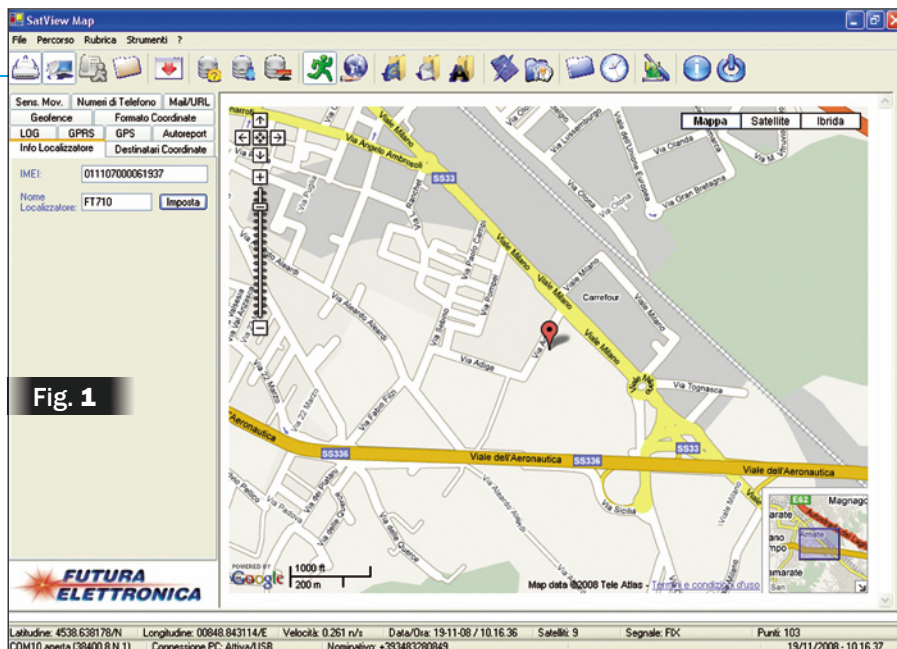


Fig. 1

telefono chiamante ed entra nel normale esercizio, dove si porta anche se il tempo passa senza che giunga alcuna chiamata.

REALIZZAZIONE PRATICA

Bene, diciamo pure che per quanto riguarda la teoria non c'è altro da aggiungere; possiamo perciò occuparci degli aspetti pratici, ossia della costruzione del localizzatore, per la quale è previsto un piccolissimo circuito stampato sul quale prenderanno posto tutti i componenti, ad eccezione del modulo GPS. La traccia lato rame dalla quale ricavare la pellicola potete scaricarla (insieme al file gerber corrispondente) dalla sezione *download* del nostro sito Web www.elettronica.in.it.

Viste le dimensioni estremamente ridotte del tutto, i componenti adottati sono in SMD, quindi il montaggio deve essere condotto con estrema attenzione, utilizzando un saldatore di piccola potenza (20 watt) con punta molto sottile e filo in lega saldante del diametro di 0,5 mm. Per il modulo GSM/GPRS è previsto un piccolo connettore ad alta densità da stagnare sullo stampato, magari aiutandosi con una lente di ingrandimento per vedere bene che non ci siano "baffi" di stagno a cortocircuitare piazzole vicine; inserito il modulino, su di esso

si fissa e si connette il porta-SIM. La connessione dell'antenna è, per il cellulare, un triplo contatto a molla dorato che, spingendo il componente a fondo nel suo connettore, tocca una piazzola che abbiamo previsto su un angolo del c.s.; su di essa si può stagnare il cavetto dell'antenna o, a seconda della sagoma, l'antenna stessa. Per il nostro prototipo, come antenna è stato usato un semplice spezzone di filo lungo circa 10 cm, tuttavia, a seconda dello spazio disponibile, si può optare per un'antenna specifica da cellulare, magari anche di quelle realizzate su stampato. Il ricevitore GPS sta al di fuori dello stampato e si collega mediante un apposito cavetto a otto poli. Per la connessione con il computer, bisogna montare sul circuito stampato un connettore mini-USB. Completato il montaggio, attraverso il connettore CN1 (8 poli) si può provvedere alla programmazione in-circuit del microcontrollore. Poi, il localizzatore è pronto per l'uso.

IL SOFTWARE DA PC

Bene, adesso spendiamo qualche parola sul programma realizzato dalla ditta Futura Elettronica per gestire il localizzatore direttamente da computer: si chiama *Sat View Map* e prevede funzioni

utilizzabili quando il localizzatore è collegato sia fisicamente al computer (mediante USB) sia virtualmente mediante una connessione dati via modem GSM. Tipicamente quando il localizzatore è connesso direttamente al PC le funzioni disponibili sono la configurazione di tutte le funzionalità del localizzatore (esattamente come si farebbe con i comandi SMS già descritti) e lo scarico della memoria mirato a vedere le posizioni e il tracciato su una mappa stradale. Invece, per quanto riguarda la connessione tramite cellulare o modem GSM/GPRS in modalità dati, si usa tipicamente per ricevere in tempo reale i dati sulla posizione del localizzatore stesso, ma anche per modificare a distanza tutte le configurazioni.

Il programma è rilasciato sotto forma di file eseguibile e si installa facendo clic sull'icona *setup.exe*; al termine della procedura viene automaticamente creata una voce nel menu Programmi (*Futura Elettronica/Sat View Map*) facendo clic sulla quale si avvia il software.

All'avvio si accede alla finestra principale (Fig. 1) che rappresenta l'interfaccia utente; in alto si trova la barra dei menu, che sono *File* (contiene i comandi di uscita, ripetuto dall'ultimo pulsante della barra degli strumenti, e gestione file) *Percorso* (importa ed esporta dati) *Rubrica* (modifica e impostazione numero da chiama-

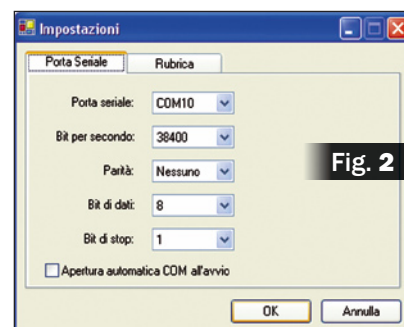
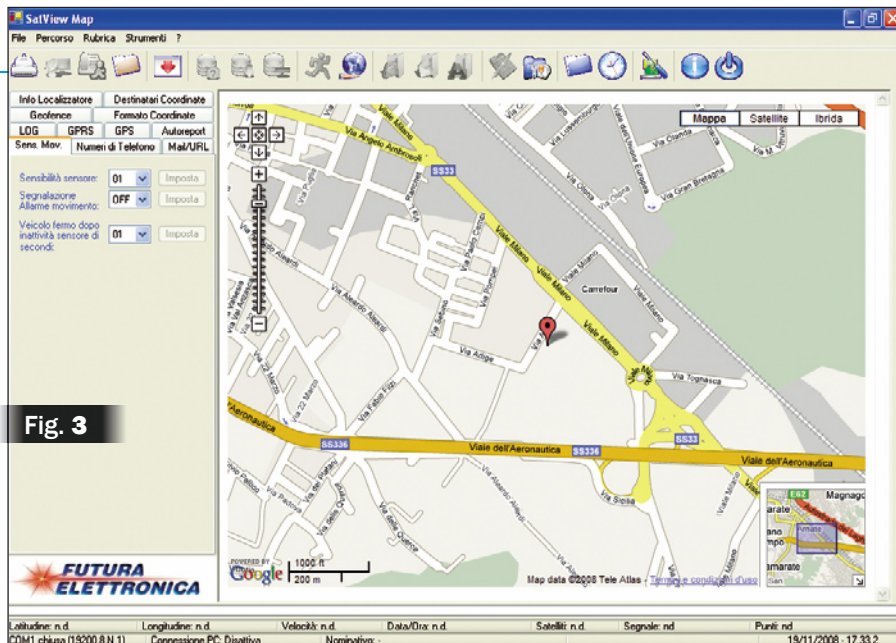


Fig. 2

re per effettuare la connessione punto-punto) e *Strumenti* (che contiene le impostazioni del fuso orario, della porta seriale virtuale relativa all'USB e altre impostazioni). I comandi del menu ed altri sono accessibili direttamente mediante i pulsanti che si trovano nella barra degli strumenti, posta immediatamente sotto quella dei menu.

Più in basso, a destra troviamo il riquadro che mostra la mappa e il punto o il percorso del localizzatore e che serve sia nelle funzioni di localizzazione real-time che in quelle di visualizzazione del tracciato scaricato dalla memoria. A sinistra, ci sono le schede che riepilogano le funzioni disponibili, ossia i comandi riguardanti la gestione del localizzatore, che ripetono quelli da SMS. Vediamo gli aspetti più interessanti, iniziando con la barra degli strumenti: da sinistra troviamo i pulsanti per l'apertura della COM virtuale assegnata al collegamento del localizzatore sull'USB, quello di connessione PC (collega il localizzatore una volta che è stata definita la COM) e quello che effettua la chiamata dati per la connessione punto/punto; facendovi clic, il modem GSM collegato al computer chiama il localizzatore per scaricare i dati della sua memoria, richiedere la posizione attuale o iniziare la connessione real-time, dove il programma visualizza nella mappa la posizione aggiornata. Quando la connessione è attiva, facendo clic sul predetto pulsante la si interrompe. Per definire il numero che il modem deve chiamare, basta fare clic sul quarto pulsante da sinistra, che dà accesso a un'apposita finestra di dialogo. Il pulsante a destra di quello che attiva/disattiva la modalità real-time serve a visualizzare la mappa del real-time.

Fig. 3



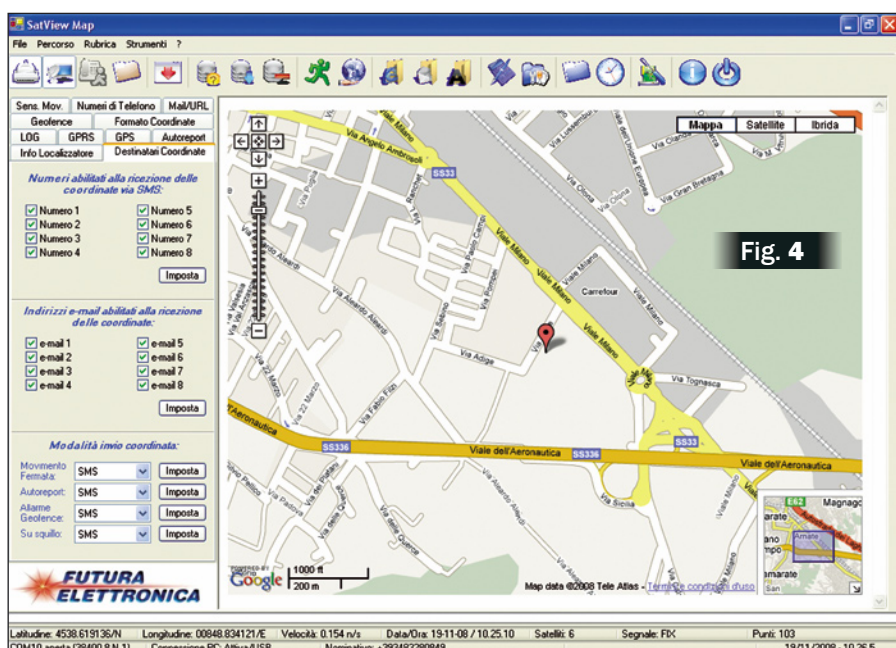
A riguardo dobbiamo far notare che in realtà quando si attiva la connessione al PC o si effettua la chiamata, il programma parte nel modo real-time, come appare evidenziato dall'attivazione del pulsante corrispondente (il nono da sinistra, cioè quello con l'uomo in corsa); per disattivare il modo real-time bisogna fare clic su tale pulsante e attendere che torni a riposo.

Notate ancora un dettaglio: ogniqualevolta si fa clic sul primo pulsante, se precedentemente era attiva una connessione si attiverà automaticamente il pulsante

corrispondente: ad esempio, se il comando si dà quando è attiva la connessione con modem, sortisce l'effetto di riattivare come dal principio la connessione. L'utilità si apprezza, ad esempio, proprio quando il modem si blocca o c'è un problema nella comunicazione in corso e, pur avendo fatto clic sul terzo pulsante per chiudere la comunicazione dati, la linea non si sconnette; in questo caso, quando il pulsante torna attivo è possibile ritentare.

Durante una connessione, i pulsanti dal sesto all'ottavo permettono, rispettivamente, di

Fig. 4



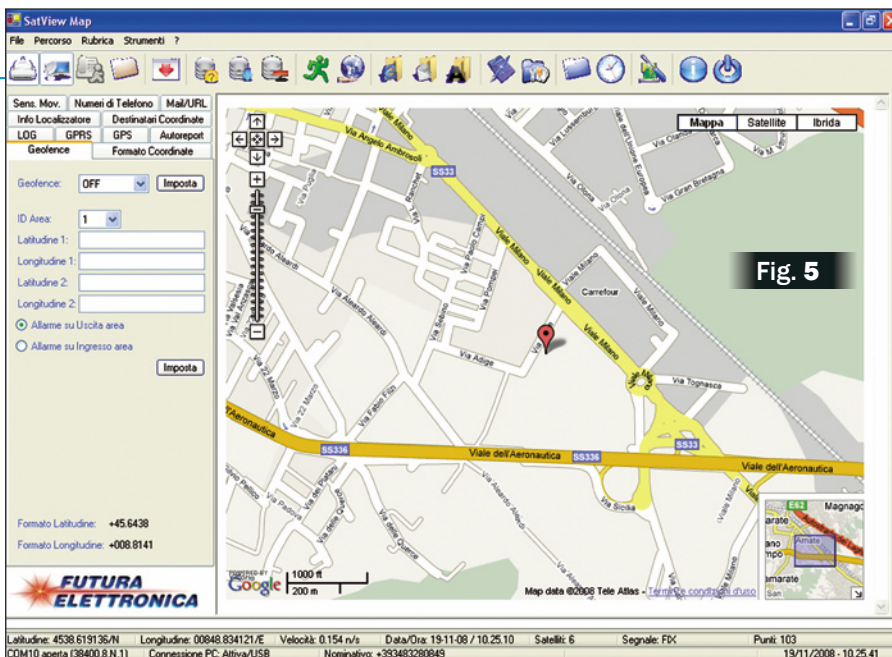


Fig. 5

verificare quanti punti il localizzatore ha attualmente in memoria (ciò serve a valutare se sia il caso di cancellare la memoria) di scaricare la memoria (download, che viene eseguito in un file sul computer) e di cancellare la memoria dei punti. Una volta effettuato un download, facendo clic sul quinto pulsante (*Importa File Coordinate*) il software carica i dati dal predetto file e ne mostra i punti sulla mappa nell'apposito riquadro della finestra principale. L'undicesimo, dodicesimo e tredicesimo pulsante servono per il test, rispettivamente, dell'URL, del servizio e-mail e dell'invio di SMS; facendovi clic, il programma chiede al localizzatore l'invio di dati su URL, di un'e-mail o di un SMS di prova. Tali pulsanti risultano attivi e utilizzabili solo quando il localizzatore è collegato direttamente all'USB del PC. Il quattordicesimo e quindicesimo pulsante permettono rispettivamente di leggere le impostazioni del localizzatore e scaricare il file di configurazione corrispondente; con il primo la sezione di sinistra (tutte le schede) viene compilata automaticamente con le attuali impostazioni del localizzatore attualmente collegato via USB o GSM, impostazioni che il programma salva in un apposito file.

Il secondo è un po' più particolare, perché permette di richiamare l'ultimo file di configurazione scaricato da un localizzatore, anche se non più collegato al computer; serve essenzialmente quando si desidera impostare un nuovo localizzatore come l'ultimo dal quale è stato letto il file di configurazione. Facendo clic sul pulsante 15, il programma carica il file e aggiorna le schede; per applicare le impostazioni basta, scheda per scheda, sezione per sezione, fare clic sul pulsante *Imposta*. Ad ogni pressione, il localizzatore riceve i dati cor-

rispondenti e li memorizza. Il sedicesimo pulsante visualizza la rubrica, ossia la lista dei numeri telefonici, mentre il diciassettesimo permette di impostare il fuso orario. Quest'ultima funzione serve a fare in modo che il sistema visualizzi l'orario corrente nel luogo dove si trova, altrimenti l'ora considerata è quella di Greenwich (un'ora in meno di quella italiana).

Il diciottesimo pulsante dà accesso all'omonima finestra di dialogo *Impostazioni*, che si compone delle due schede *Rubrica* e *Porta Seriale*; quanto alla rubrica, serve a memorizzare i numeri che il PC può chiamare quando si preme il terzo pulsante, ovvero quando tenta la connessione diretta via GSM. Infatti, in tal caso a video appare una finestra contenente la lista dei numeri in rubrica, dalla quale bisogna scegliere quello corrispondente alla SIM del localizzatore. La rubrica permette di memorizzare i numeri di più localizzatori e di richiamare di volta in volta quello voluto. *Porta Seriale* è, invece, la scheda dalla quale si impostano i parametri

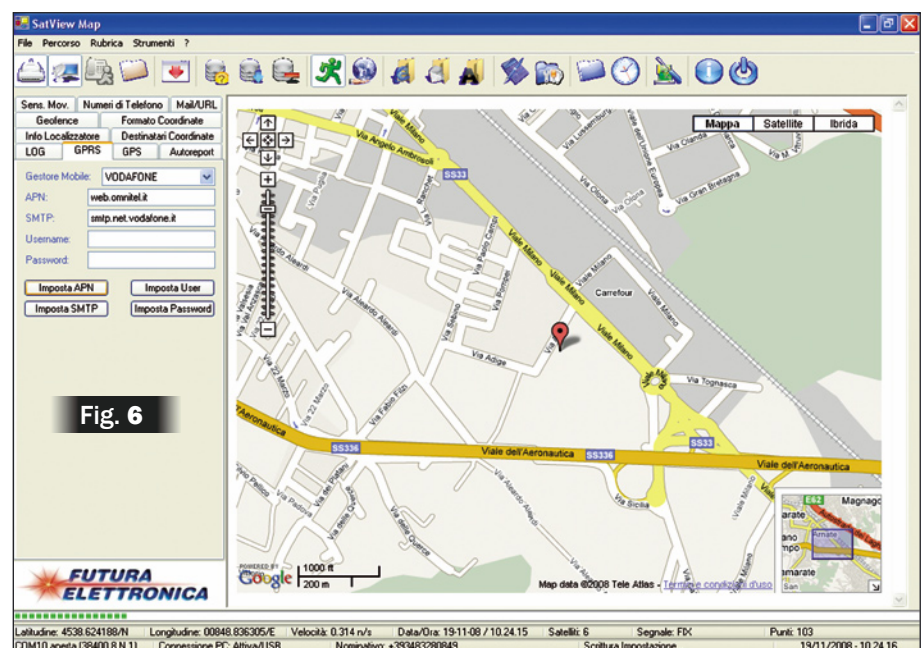


Fig. 6

della seriale (Fig. 2); in essa, spuntando la casella *Apertura automatica COM all'avvio* ogni volta che si avvia il programma viene aperta la porta prevista, senza bisogno che si intervenga sul primo pulsante della barra degli strumenti.

Descritti i pulsanti, vediamo le singole schede, con la premessa che all'avvio del programma quella aperta è *Info localizzatore*; in essa troviamo le caselle per vedere l'IMEI del cellulare e il nome eventualmente assegnato all'apparato; le caselle vengono compilate usando il pulsante 15. La scheda *Sens. Mov.* (Fig. 3) è quella che riguarda le impostazioni inerenti al sensore di movimento e permette di attivare e disattivare la funzione corrispondente, definire la sensibilità e il tempo dall'arresto trascorso il quale deve partire la segnalazione.

C'è poi la scheda *Destinatari Coordinate* (Fig. 4) che riepiloga gli indirizzi di posta e i numeri telefonici della lista cui il localizzatore dovrà dirigere e-mail ed SMS con le coordinate; permette inoltre di modificare tali dati e definire quando inviare le coordinate stesse. *Geofence* (Fig. 5) è la scheda con cui si definisce l'omonima funzione: nelle caselle Latitudine 1, Longitudine 1, Latitudine 2 e Longitudine 2 si scrivono latitudine e longitudine che definiscono le zone previste; le ultime due caselle consentono di stabilire se l'allarme debba scattare uscendo dall'area oppure entrandovi.

La scheda *GPRS* (Fig. 6) serve a definire i parametri del servizio dati, ovvero il gestore, l'APN (Access point Name) ossia il punto di accesso dalla rete cellulare a Internet, il server di posta in uscita e username e password, laddove sia richiesta l'autentica-

zione dell'utente.

Mail/URL (Fig. 7) è la scheda dove si definiscono gli otto possibili indirizzi di posta elettronica cui inviare e-mail contenenti le coordinate, ovvero l'URL cui dirigere le stesse quando si opta per l'invio su URL.

Formato Coordinate è la scheda che contiene i comandi corrispondenti ai parametri dei comandi SMS *FORS* e *FORE*; le caselle che contiene permettono di scegliere il tipo di formato (fare riferimento a quanto spiegato per i predetti comandi da SMS).

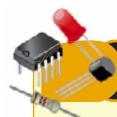
Numeri di Telefono è la scheda che riepiloga i numeri telefonici memorizzabili nel localizzatore: da essa si può sia vedere quelli attuali, sia sostituire alcuni di essi o tutti con dei nuovi; non va confusa con la rubrica, che è, invece, una lista di numeri che il PC deve chiamare e che riguardano le SIM di uno o più localizzatori.

La scheda *GPS* permette di comandare direttamente l'accensione e lo spegnimento del ricevitore; serve, come il corrispondente comando SMS, per disattivare manualmente il GPS quando si desidera riavviarlo perché le coordinate ricevute sono incoerenti oppure qualora si intenda allungare il più possibile l'autonomia del localizzatore.

Autoreport è la scheda che

permette di definire le modalità dell'autoreport, ossia gli intervalli di invio sia nel modo incondizionato che in quello asservito al movimento del veicolo.

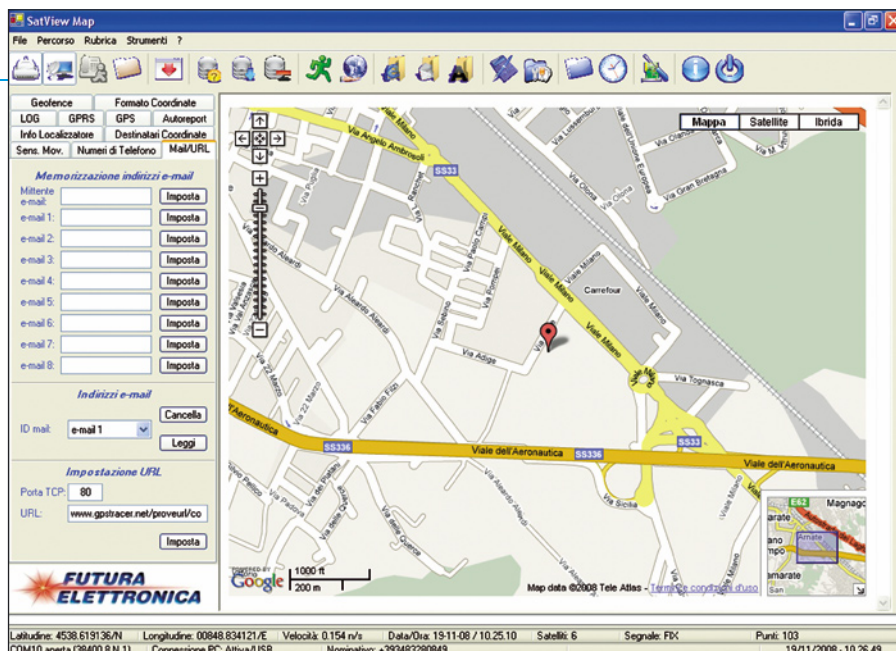
L'ultima scheda presente nella finestra principale del programma è *LOG* e permette di definire le modalità di memorizzazione delle coordinate da parte del localizzatore; in essa troviamo le caselle con cui attivare e disattivare la registrazione ciclica incondizionata (e la rispettiva periodicità) e quella a seguito di evento (anche in questo caso possiamo definire la cadenza).



per il MATERIALE

Il localizzatore è disponibile già montato e collaudato al prezzo di 275,00 Euro (cod. FT710M). Il dispositivo comprende il modulo principale con tutti i componenti, la batteria al litio, il ricevitore GPS a 32 canali FV-M8, il software di gestione nonché un completo manuale illustrato. Non è compreso alcun contenitore. L'alimentatore FT681M-36 (necessario per alimentare il localizzatore con la batteria auto) è disponibile al prezzo di 19,00 Euro. Tutti i prezzi si intendono IVA compresa.

Il materiale va richiesto a:
Futura Elettronica, Via Adige 11,
21013 Gallarate (VA)
Tel: 0331-799775 • Fax: 0331-792287
<http://www.futurashop.it>





CORSO ZigBee[®]

in collaborazione con
AUR·EL
di FRANCESCO FICILI

Si sta facendo strada nel panorama dei dispositivi per reti wireless a breve distanza: è ZigBee, un protocollo di comunicazione che ha destato l'interesse di tanti costruttori. Scopriamolo insieme. Prima puntata.

Con il termine ZigBee viene definito il protocollo di comunicazione di alto livello, basato sullo standard IEEE 802.15.4, creato per l'implementazione e la standardizzazione di reti Wireless PAN (Personal Area Network) a bassa velocità di comunicazione. Il protocollo ZigBee (a volte indicato anche come stack ZigBee, o standard ZigBee) prevede lo scambio di dati nelle frequenze radio assegnate per scopi industriali, scientifici e medici (indicate genericamente con il termine ISM) a 433 o 868 MHz in Europa, 315 o

915 MHz negli Stati Uniti e 2,4 GHz a diffusione praticamente mondiale. ZigBee nasce come alternativa, più semplice ed economica, ad altri tipi preesistenti di standard per WPANs, come, per citarne due tra i più diffusi, Bluetooth e Wi-Fi. Lo standard prevede comunicazioni a bassa velocità, realizzate con trasmettitori di bassa potenza, oltre alla creazione di topologie di rete anche molto complesse e ad elevato numero di nodi. Una delle caratteristiche peculiari delle WPAN ZigBee è la possibilità di garantire la comunicazione tra nodi



che non si “vedono” direttamente, sfruttando il mesh networking. Tipici target applicativi sono le applicazioni embedded che richiedano transfer-rate non elevati e consumi ridotti. La finalità dello standard è quella di definire una tipologia di rete wireless non mirata, economica e autogestita, che possa essere utilizzata per scopi quali il monitoraggio e il controllo industriale, le reti di sensori, la domotica, ecc.

L'aggiornamento, la revisione e la pubblicazione dello standard, sono affidati alla ZigBee Alliance, un consorzio di aziende (il cui logo è illustrato nella Fig. 1) del settore telecomunicazioni ed elettronica. La specifica ZigBee 1.0 è stata approvata il 14 dicembre 2004 ed attualmente la ZigBee Alliance ha iniziato a lavorare sulla versione 1.1. I membri della ZigBee Alliance sono divisi in tre livelli: Promoters, Participants ed Adopters. Attualmente l'adesione a livello Adopters costa a un'azienda circa 3500 dollari all'anno e permette l'accesso alle specifiche e al relativo uso per scopi commerciali. Al giorno d'oggi si contano circa 280 aziende affiliate alla ZigBee Alliance e tra i promotori figurano società del calibro di ST Microelectronics, Samsung, Texas Instruments, Philips, Motorola, Freescale, Siemens ed altre ancora. Tra i Participants figurano aziende del

ZigBee Specification

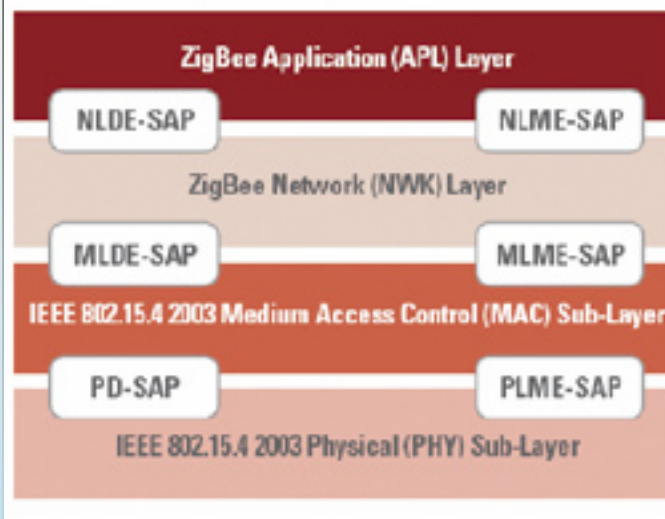


Fig. 2 - Rappresentazione dello standard ZigBee secondo il modello ISO/OSI

settore elettronico e produttori di microcontrollori come Microchip Technology, Atmel, Fujitsu, Renesas, Telit, ma anche aziende di settori non strettamente connessi con quello delle telecomunicazioni, come ad esempio FedEx e Black&Decker.

DESCRIZIONE DELLO STANDARD

Per analizzare lo standard ZigBee partiamo dalla sua “carta d'identità”, ovvero, parlando in termini di protocolli di comunicazione, dalla sua rappresentazione secondo il modello ISO/OSI, una cui schematizzazione semplificata è visibile nella Fig. 2.

Come si vede, il livello Fisico ed il livello MAC fanno parte della specifica 802.15.4; la specifica ZigBee aggiunge altri due livelli, che sono un livello Network (NWK) ed un livello

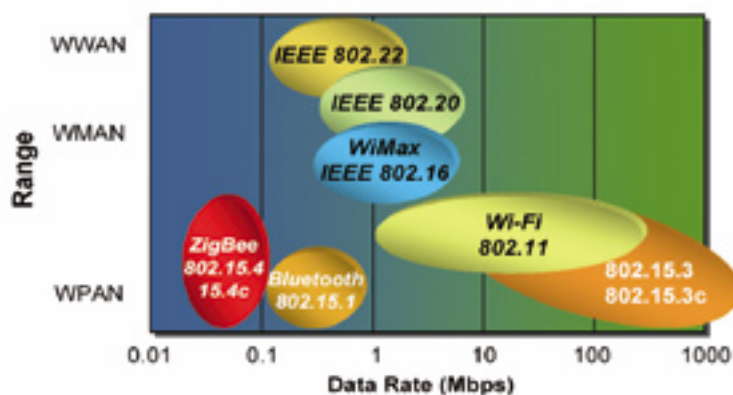
Application (APL). I riquadri in bianco identificano i SAP (Service Access Point), che costituiscono i punti di accesso e comunicazione tra due layer (livelli o strati) contigui.

Prima di addentrarsi nella descrizione dei vari strati che lo compongono, è opportuno fare un piccolo inciso sulle tipologie di dispositivi ammessi dallo standard ZigBee, in modo da poter meglio comprendere i paragrafi che seguiranno. Lo standard ZigBee ammette tre differenti tipologie di dispositivi.



Fig. 1 - Logo della ZigBee Alliance.

Confronto tra le più diffuse tecnologie wireless



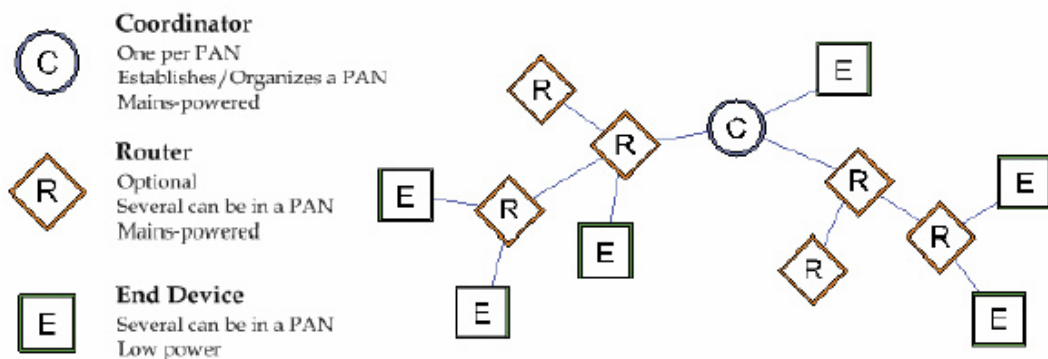


Fig. 3
Esempio di rete ZigBee.

- **ZigBee Coordinator:** è il dispositivo più "intelligente" tra quelli disponibili, costituisce la radice di una rete ZigBee e può operare da ponte tra più reti. Ci può essere un solo Coordinator in ogni rete. Assolve funzioni vitali, come la creazione della rete stessa, la selezione del canale RF e l'assegnazione degli identificativi. È, inoltre, in grado di memorizzare informazioni sulla rete creata e può agire come deposito per le chiavi di sicurezza. Un malfunzionamento di questo dispositivo può ripercuotersi sul resto della rete o addirittura causarne il blocco.
- **ZigBee Router:** agisce come router intermedio, passando i dati da e verso altri dispositivi. In una PAN può essercene più di uno; esso dispone di un certo livello di intelligenza, in quanto implementa complicati algoritmi di routine per la selezione dei percorsi migliori da e verso un determinato nodo. Un malfunzionamento di uno di questi dispositivi non pregiudica il funzionamento di tutta la rete, ma può renderne inutilizzabile una parte.
- **ZigBee End Device:** questo dispositivo include solo le funzionalità minime per dialogare con il loro nodo parente (Coordinator o Router); può essercene più di uno e non può trasmettere dati provenienti da altri dispositivi. I ZigBee End Device sono i nodi che richiedono il minor quantitativo di memoria e quindi risultano spesso più economici rispetto ai Router ed ai Coordinator.

Un esempio di una semplice rete ZigBee nella quale sono presenti tutti e tre i tipi di dispositivi è illustrato nella Fig. 3.

Passiamo ora alla descrizione dei vari strati che compongono lo stack ZigBee, partendo dal basso verso l'alto e servendoci della Fig. 4, che non è altro se non una rappresentazione più dettagliata, nella quale sono evidenziati anche interfacce e sublayer, dell'architettura dello stack ZigBee.

Livello Fisico

Il livello fisico è definito dallo standard IEEE 802.15.4, ed ha il compito di fornire l'interfaccia tra il canale trasmissivo ed il livello MAC dello standard. Secondo le specifiche, il livello fisico deve essere in grado di fornire i seguenti servizi:

- attivazione e disattivazione del transceiver;
- rilevamento dell'energia del canale in uso;
- rilevamento della qualità del collegamento;

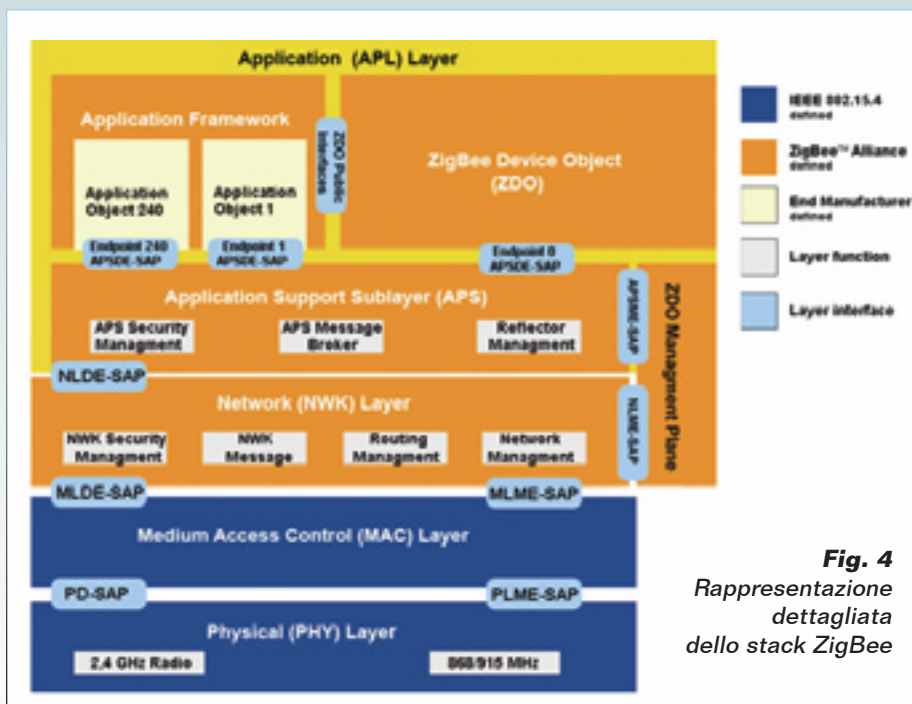


Fig. 4
Rappresentazione dettagliata dello stack ZigBee

	Frequenza operativa [GHz]	Potenza RF	Interfaccia di comunicazione	Antenne disponibili	Package	Note
Aurel	2,4	1mW (moduli xLI) 20mW (moduli xHI)	UART	Chip o esterna	Pin Strip passo 2mm (modulo)	Software demo dedicato. OCX dedicato
Microchip	2,4	1,26mW	SPI	Chip	QFN 40 pin	Supporto per 802.15.4 MAC e AES encryption
Digi	2,4	1mW (moduli standard) 100mW (moduli PRO)	UART	Filo, chip, U.FL e RPSMA	Pin Strip passo 2mm (modulo)	Supportate modalità di rete avanzate e a basso consumo
Atmel	2,4	2mW	SPI	Chip	QFN 40 pin	
Jennic	2,4	3mW	UART, SPI	Chip	QFN 56 pin	Supporto per 802.15.4 e AES encryption

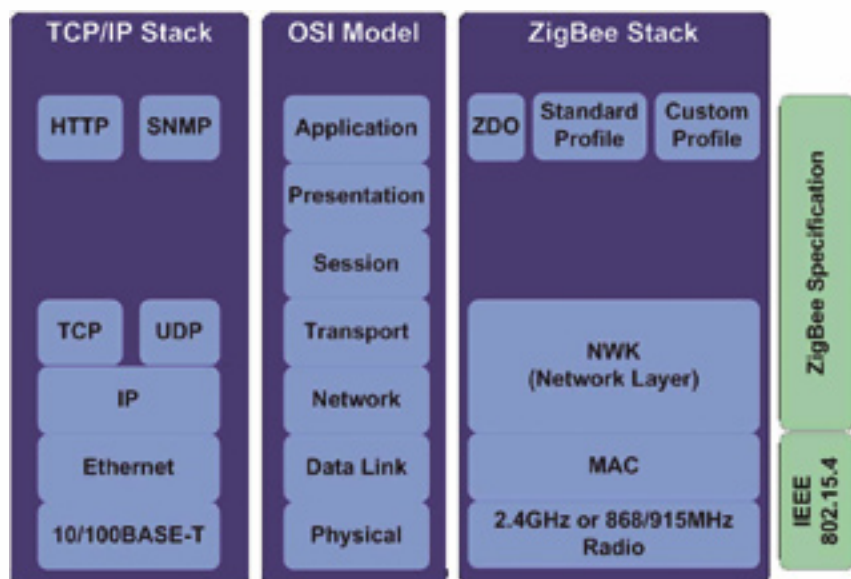
Tabella 1
Cross
reference
Moduli
ZigBee

- selezione della frequenza di comunicazione;
- fornitura dei servizi per l'accesso al canale trasmissivo con la tecnica del CSMA/CA;
- trasmissione e ricezione dei dati.

Il livello fisico è concettualmente costituito da due entità distinte, la prima delle quali è un'entità di comando generalmente chiamata PHY Layer Management Entity (PLME) che costituisce la vera e propria interfaccia con il livello MAC; la seconda è invece un'entità deputata esclusivamente al trasferimento dei dati e viene generalmente indicata con il nome di PHY Layer Data Entity Service (PLDE). Entrambe le entità possiedono un Service Access Point dedicato, che viene utilizzato dal livello MAC per gestire l'intero livello Fisico e comunicare con esso; i due SAP

sono rispettivamente il PHY Layer Management Entity Service Access Point (PLME-SAP) ed il PHY Data Service Access Point (PD-SAP). Un discorso a parte lo merita la tecnica di accesso al canale trasmissivo, che secondo le specifiche deve essere di tipo CSMA/CA. CSMA/CA è l'acronimo inglese di Carrier Sense Multiple Access with Collision Avoidance, ovvero accesso multiplo tramite ascolto del canale con elusione delle collisioni. Nel momento in cui una stazione vuole tentare una trasmissione, ascolta il canale (Listen-before-Transmit). Se lo trova occupato, la stazione attiva un timer di durata casuale (detto tempo di backoff) che viene decrementato solo durante i periodi di inattività del canale. Quando il timer scatta, la stazione fa un altro tentativo.

Rappresentazione OSI degli stack Ethernet e ZigBee



Ciò permette di evitare, su base statistica, che più nodi tentino di accedere al canale contemporaneamente.

Si è scelto di utilizzare questa tecnica perché nelle reti wireless non è possibile rilevare le collisioni, come si fa nel CSMA/CD (Carrier Sense Multiple Access with Collision Detection), motivo per cui le collisioni devono essere evitate piuttosto che rilevate.

Livello MAC

Anche il livello Medium Access Control (MAC) come il livello Fisico, è definito dallo standard IEEE 802.15.4 e secondo le specifiche deve essere in grado di fornire i seguenti servizi:

- supporto all'avvio ed al man-

- tenimento della PAN;
- indirizzamento dei dispositivi;
- esecuzione dell'algoritmo CSMA\CA per l'accesso al canale trasmissivo;
- allocazione dei GTS (Guaranteed Time Slot);
- supporto alla sicurezza dei pacchetti trasmessi.

Come quello Fisico, anche il livello MAC può essere concettualmente diviso in due entità, ossia il MAC Sublayer Management Entity (MLME), che fornisce il servizio di gestione del livello e delle interazioni con il resto dello stack (come il supporto alla creazione di una PAN) ed il MAC Sublayer Data Entity Service (MLDE) che si occupa, invece, del trasporto dei dati. I servizi forniti sono accessibili da due SAP: all'MLDE si accede attraverso il MAC Common Part Sublayer Service Access Point (MCPS-SAP), mentre l'MLME è accessibile tramite il MAC Sublayer Management Entity Service Access Point (MLME-SAP). Relativamente all'indirizzamento, è importante far notare che ogni dispositivo ZigBee è univocamente identificato da un numero di serie a 64 bit, che viene scritto in una ROM interna al chip e che permette così di distinguerlo in tutto il mondo da qualsiasi altro dispositivo della stessa famiglia.

Livello Network

Il livello Network (NWK) del protocollo è definito direttamente dalla ZigBee Alliance. I suoi compiti sono di eseguire il routing dei pacchetti dalla sorgente alla destinazione, eseguire l'associazione e la dissociazione di un nodo da una network, applicare gli algoritmi di sicurezza e, nei coordinatori, inizializzare la rete ed assegnare gli indirizzi. Anche il livello Network, come i precedenti, è formato da due entità: la Network Layer Data Entity (NLDE) e la Network Layer Management Entity (NLME). La NLDE fornisce il servizio di trasporto dei pacchetti ed applica la

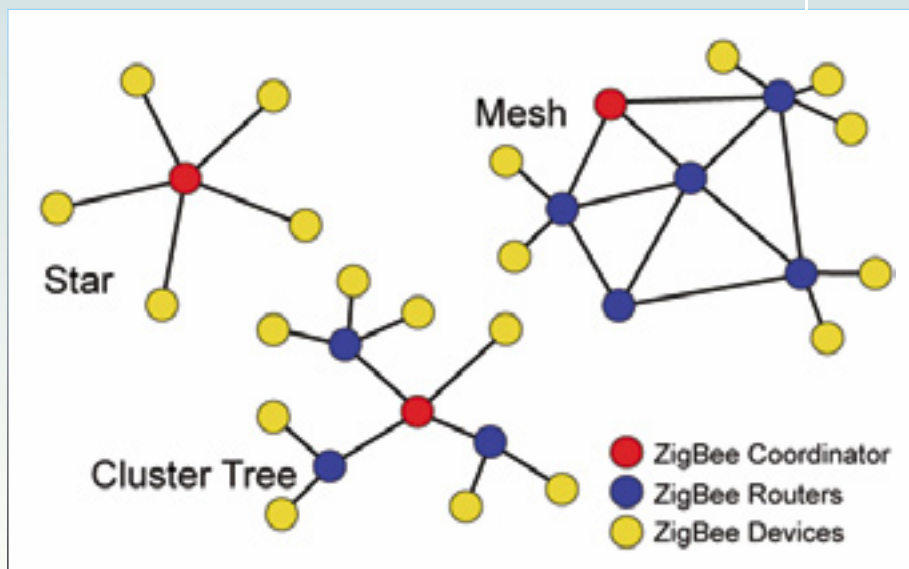
sicurezza; i suoi compiti sono essenzialmente la generazione dei pacchetti dati a livello Network, il routing e l'applicazione della sicurezza sui dati. Il livello Application può accedere ai servizi forniti dalla NLME attraverso il Network Layer Data Entity Service Access Point (NLME-SAP). La NLME fornisce invece servizi volti a garantire ad una applicazione la possibilità di interagire correttamente con lo stack ZigBee. La fornitura di questi servizi al livello superiore è garantita dal Network Layer Management Entity Service Access Point (NLME-SAP).

A livello Network sono definite essenzialmente tre diverse topologie di rete: Star, Mesh e Cluster Tree. La Star Network è costituita attorno ad un PAN Coordinator al quale è collegato un numero variabile di dispositivi figli, i quali possono essere sia Router che End Device. Nel caso di topologia cluster tree il coordinatore è collegato a stella ad una serie di Router ed End Device; i Router, a loro volta, ripropongono una topologia a stella. Infine, nella topologia Mesh ogni nodo è collegato a tutti quelli che si trovano nel suo raggio di copertura. Questa topologia viene anche chiamata rete peer-to-peer. Nella Fig. 5 sono rappresentate graficamente le tre topologie di rete previste dallo standard.

Livello Application

Il livello Application dello standard viene definito direttamente dalla ZigBee Alliance.

Fig. 5
Topologie
di rete
previste
dallo
standard
ZigBee.



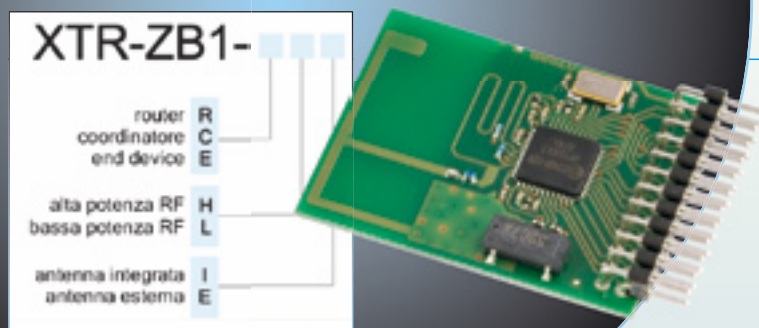


Fig. 6 Modulo ZigBee prodotto da Aurel.

Il layer corrispondente è formato dall'Application Support Sublayer

(APS), dallo ZigBee Device Object (ZDO) e dagli Application Objects implementati dal costruttore del chip in questione.

L'Application Support Sublayer fornisce due interfacce, che sono l'APS Management Entity Service Access Point (APSME-SAP) e l'APS Data Entity Service Access Point (APSDESAP). La prima di queste interfacce viene utilizzata per applicare la sicurezza, mentre la seconda è usata dagli Application Objects e dagli ZDO per la trasmissione dei dati.

Gli Application Object sono definiti dal costruttore del chip e costituiscono le applicazioni utente che si ritrovano al livello più alto dello stack; risiedono su endpoint numerati da 1 a 240. Per essere conformi allo standard, devono aderire a determinati profili approvati dalla ZigBee Alliance. Gli endpoint, insieme all'indirizzo del dispositivo, forniscono un sistema per indirizzare in maniera uniforme ed univoca su tutta la rete i singoli

Application Object (ossia le applicazioni finali) all'interno di un dispositivo. Il dualismo che esiste tra endpoint ed indirizzi ZigBee è simile a quello che viene individuato dalla coppia indirizzo IP del server/numero di porta TCP in una rete TCP/IP.

Gli ZigBee Device Objects (ZDO) forniscono un'interfaccia

verso gli Application Objects usati per scoprire altri dispositivi e servizi forniti da questi. Lo ZDO è un Application Object speciale, che risiede sull'endpoint 0.

SOLUZIONI ESISTENTI

Attualmente, per l'implementazione di reti ZigBee, esistono in commercio diverse soluzioni hardware; vediamo le caratteristiche di alcune tra le più diffuse.

Aurel

Aurel Wireless propone due moduli in grado di operare su rete standard ZigBee in maniera efficiente, sicura e con possibilità di controllo da parte del coordinatore di rete dell'integrità e funzionalità della stessa. La prima serie di moduli è individuata dalla sigla XTR-ZB1 e, come si vede nella Fig. 6, le schede sono differenziate per tipologia di dispositivo (Coordinatore, Router, End Device), per la massima potenza irradiata (High Power e Low Power) e per la tipologia di antenna (interna ed esterna) prevista.

Alcune caratteristiche di questi moduli sono: possibilità di cambio automatico del canale RF in uso in presenza di disturbi, scambio dati tipo mesh tra dispositivi della stessa rete utilizzando stringhe ASCII con indirizzo più dati, trasmissione dati codificata (128 bit di cifratura) automatica e trasparente all'utente, set di comandi ASCII per l'utilizzo a basso livello dei moduli.

Sono basati sul system-on-chip Texas Instruments cc2430, un potente chip che contiene al suo interno un transceiver ZigBee associato ad un microcontrollore 8051, con tagli di flash che arrivano fino a 128 kbyte. La quantità di memoria è sufficiente ad ospitare l'intero stack ed un'applicazione utente firmware. Nel caso specifico, Aurel ha realizzato il firmware per l'implementazione di nodi Coordinator, Router ed End Device, più un application layer che permette di gestire la UART, il convertitore A/D, ingressi ed uscite digitali, interrupt ed altro ancora. Volendo espanderne le funzionalità, può essere associato ad un qualsiasi microcontrollore dotato di porta USART.

Inoltre, per permettere un semplice approccio ai suoi moduli wireless ZigBee, Aurel mette a disposizione un completo kit di sviluppo che

Fig. 7 Starter kit ZigBee prodotto da Aurel



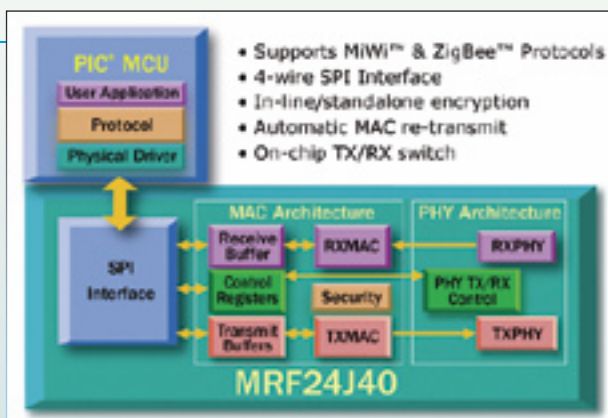


Fig. 8 - Schema a blocchi del modulo MRF24J40.

permette di testare tutte le funzionalità dei dispositivi.

Il kit (visibile nella Fig. 7) comprende una scheda USB per l'interfacciamento del PAN coordinator al PC e 4 demoboard predisposte per la connessione con altrettanti router; sono altresì inclusi tutta la documentazione necessaria, i driver dell'interfaccia USB e una serie di tool software per lo sviluppo di applicazioni.

Microchip

La Microchip's Radio Frequency Products Division, divisione di Microchip dedicata ai prodotti RF, offre diverse soluzioni per lo sviluppo di sistemi wireless a corto raggio, tutti basati sul chip MRF24J40. Questo dispositivo è un transceiver wireless a 2,4 GHz a standard IEEE 802.15.4, pensato per l'utilizzo con protocolli di alto livello come ZigBee.

Nella Fig. 8 è possibile vedere lo schema a blocchi del transceiver MRF24j40 nella sua applicazione tipica, ossia l'interfacciamento con un microcontrollore PIC di Microchip. A supporto dei progettisti, Microchip Technology fornisce una serie di stack software adatti all'implementazione su microcontrollori PIC. Le soluzioni proposte da Microchip sono le seguenti:

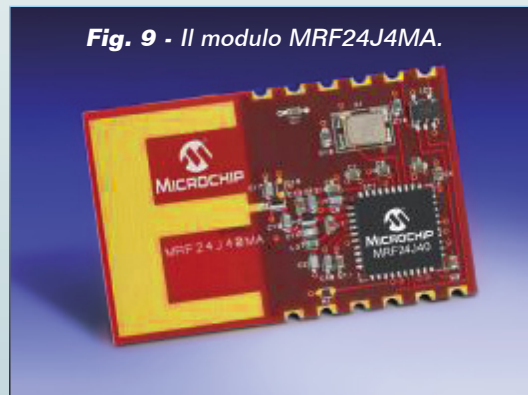
- ZigBee;
- MiWi;
- MiWi P2P.

Lo stack ZigBee ha una occupazione di memoria programma che può arrivare sino a 96 kbyte (escluso il codice utente), quindi è limitato a quei microcontrollori PIC aventi tagli di memoria adeguati. Sono supportate le topologie a stella e a mesh, mentre come target si hanno a disposizione i microcontrollori di

Tabella 2 - Caratteristiche degli stack Microchip.

Stack	Topography	Hops	Memory	PIC18	PIC24	PIC32
MiWi™ P2P (Proprietary)	Star/P2P	1	3K-9K	SI	SI	SI
MiWi (Proprietary)	Star/Mesh	4	7K-17K	SI	SI	SI
Zigbee (Certified)	Star/Mesh	Infinite	40K-96K	SI	SI	NO

Fig. 9 - Il modulo MRF24J40MA.



casa Microchip ad 8 e 16 bit (PIC24 e dsPIC); attualmente non sono supportati i PIC32. Lo stack MiWi\MiWi P2P non è altro che una versione semplificata dello stack ZigBee: il footprint del codice si riduce a 7÷17 kbyte (3÷9 Kbyte per MiWi P2P) e risulta quindi adeguato per applicazioni cost-sensitive, che utilizzano microcontrollori con limitate quantità di memoria flash. Sono supportati microcontrollori PIC18, PIC24, dsPIC30/33 e anche i recenti PIC32.

Nella Tabella 2 sono elencate le caratteristiche più importanti dei tre stack.

Il chip MRF24j40 si presenta con un package di tipo QFN a 40 pin, quindi poco adatto alla prototipazione; ecco perché Microchip ha messo in commercio un modulo RF basato su questo chip, con già a bordo tutta l'elettronica necessaria al funzionamento (antenna inclusa) e un agevole collegamento verso l'esterno. Il modulo prende il nome di MRF24j40MA ed è raffigurato nella Fig. 9.

Digi-Maxstream

La Digi-Maxstream, azienda specializzata nella realizzazione di soluzioni embedded, ha sviluppato una serie di moduli radio compatibili con le specifiche dello standard ZigBee. I moduli radio, che sono stati battezzati con il nome di Xbee RF modules, presentano una forma adatta alla prototipazione (due file di pin strip a passo 2 mm) e Digi mette a disposi-

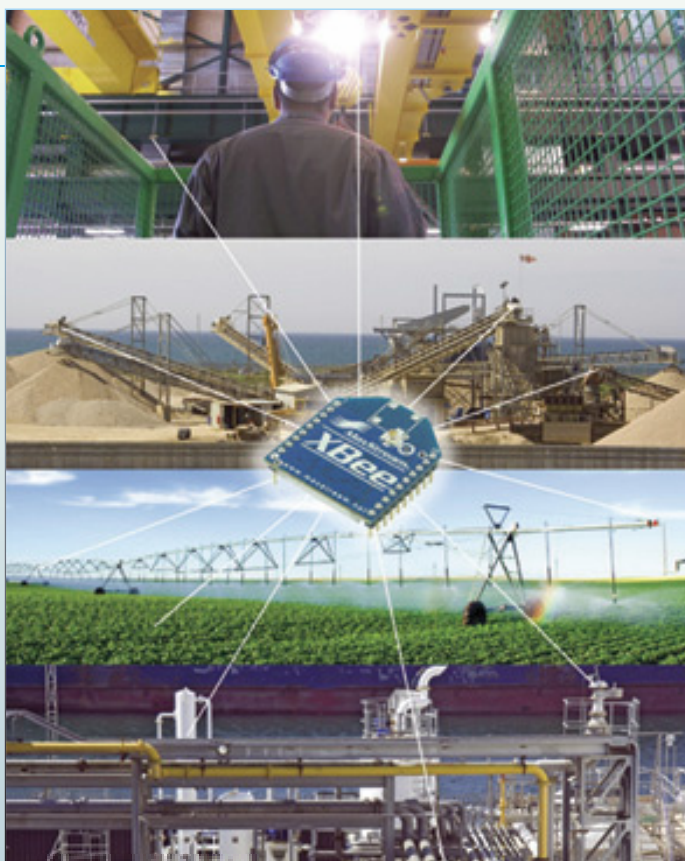


Fig. 10 Scenario applicativo dei moduli Xbee prodotti da Digi.

denominata X-CTU, attraverso cui è possibile programmare i moduli, effettuare il range test e dialogare con i moduli stessi attraverso un terminale dotato di interfaccia seriale.

Sono inoltre a disposizione varie versioni di firmware per l'implementazione di reti ZigBee. In particolare, è possibile realizzare sistemi di comunicazione wireless sia point-to-point che mesh. Per la soluzione point-to-point possono essere utilizzati i moduli denominati "serie 1", più economici ma con prestazioni inferiori. Se invece si vuole realizzare una mesh, è necessario utilizzare i moduli serie 2 (che sono in grado di supportare i nuovi firmware per implementazione di ZigBee mesh, Znet 2.5 e ZB).

Atmel

Anche Atmel, parallelamente alla concorrenza, propone una sua soluzione wireless ZigBee. In questo caso i moduli prendono il nome di Z-Link e sono ottimizzati per l'utilizzo con piattaforme di tipo AVR.

Nella Fig. 12 è riportato un possibile schema a blocchi del sistema.

Un ottimo strumento per lo sviluppo di sistemi wireless basati sulla soluzione proposta da Atmel è lo starter kit RZ200, visibile nella Fig. 13.

Jennic

La Jennic è un'azienda specializzata nella produzione di soluzioni wireless on-chip; la sua vasta esperienza nel settore della connettività senza fili ha portato alla realizzazione di



zione una ricca dotazione di librerie firmware sia per soluzioni point-to-point che per l'implementazione di reti mesh.

La soluzione proposta da Digi presenta caratteristiche interessanti per gli hobbisti, in quanto i moduli possono essere facilmente montati su schede elettroniche per prototipazione o PCB. Il passo dei pin strip per la connessione del modulo non è standard, ma esistono in commercio delle schede di adattamento. L'interfaccia di comunicazione è di tipo seriale UART, quindi facilmente interfacciabile ad un qualsiasi microcontrollore (PIC, AVR, ARM, ecc).

Uno dei pregi di questa soluzione è rappresentato dalle varie possibilità nella scelta dell'antenna (Fig. 11).

Per quanto riguarda il supporto offerto ai progettisti, Digi mette a disposizione un'interfaccia software per la programmazione ed il debug,

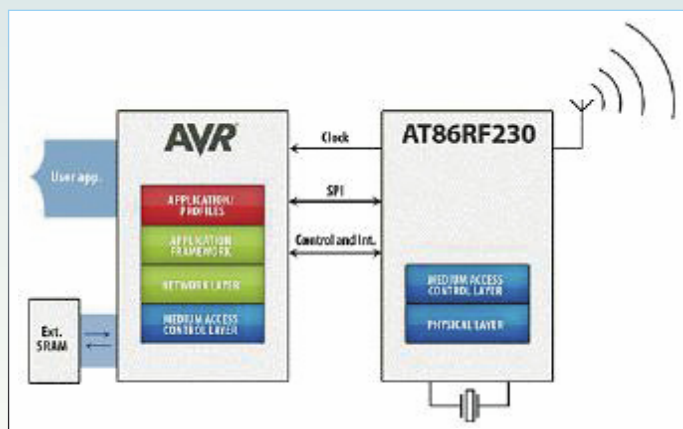
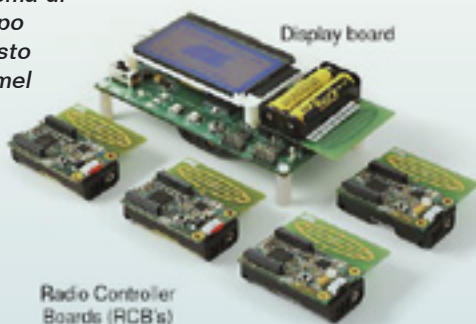


Fig. 12 - Schema a blocchi del modulo AT86RF230.

Fig. 13
Il sistema di sviluppo proposto da Atmel



un microcontrollore dotato di interfaccia IEEE 802.15.4, e quindi adatto a supportare lo stack ZigBee.

Nella Fig. 14 è rappresentato lo schema a blocchi del JN5121.

Questo versatile chip integra un microcontrollore a 32 bit ed una sezione di radiofrequenza operante a 2,4 GHz, fino a 96 kbyte di RAM e 64 kbyte di Flash. La dotazione di periferiche include Timer, ADC, 21 GPIO ed interfacce di comunicazione UART ed SPI.

La soluzione proposta da Jennic fornisce un completo sistema on-chip e può essere una buona scelta nel caso di applicazioni cost-sensitive che non richiedano prestazioni particolarmente spinte.

Un ottimo strumento per iniziare ad utilizzare i moduli realizzati dalla Jennic è l'IEEE 802.15.4 & ZigBee Evaluation Kit, rappresentato nella Fig 15.

Concludiamo qui la prima puntata del corso, abbiamo fatto una panoramica dello stack ZigBee ed analizzato alcune tra le soluzioni più interessanti che propone il mercato. Come si è visto, nonostante lo standard sia di recente concezione, per la sua implementazione in commercio esistono già diverse soluzioni, ognuna con caratteristiche proprie, vantaggi e svantaggi.

Nella prossima puntata concentreremo la nostra attenzione sul kit di sviluppo Aurel, analizzando in dettaglio i vari elementi hardware che lo compongono ed il software demo tramite il quale vi verrà fornita la possibilità di gestire la vostra prima rete ZigBee. ■

Fig. 14 - Schema a blocchi del JN5121

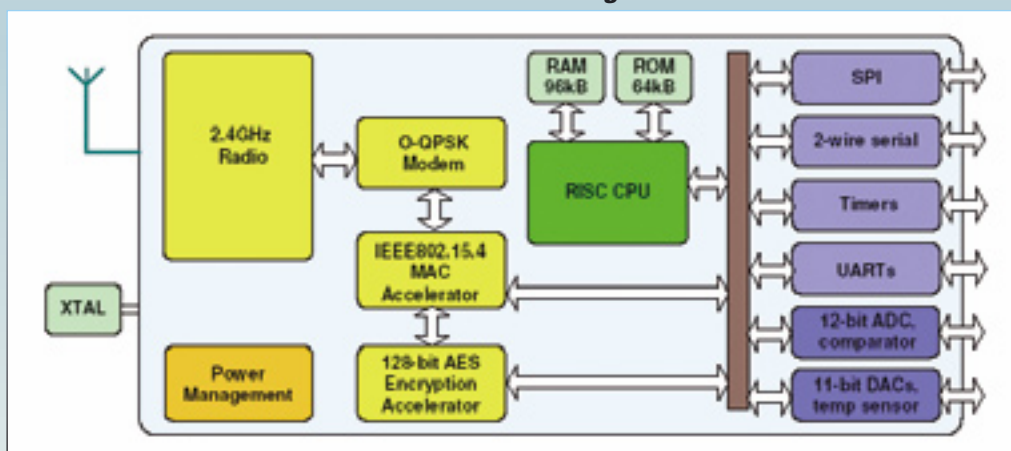


Fig. 15
Evaluation kit della Jennic.

